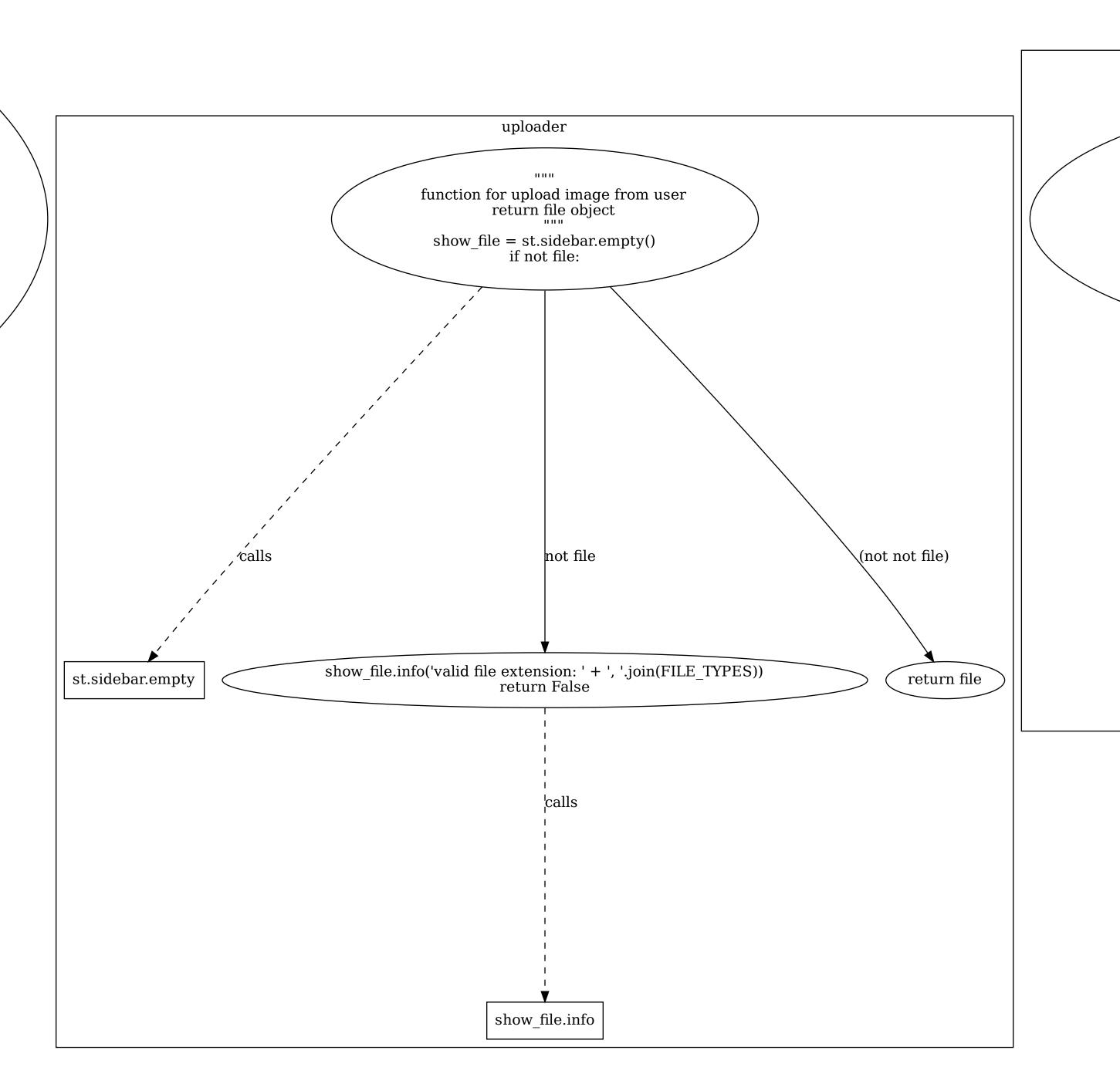
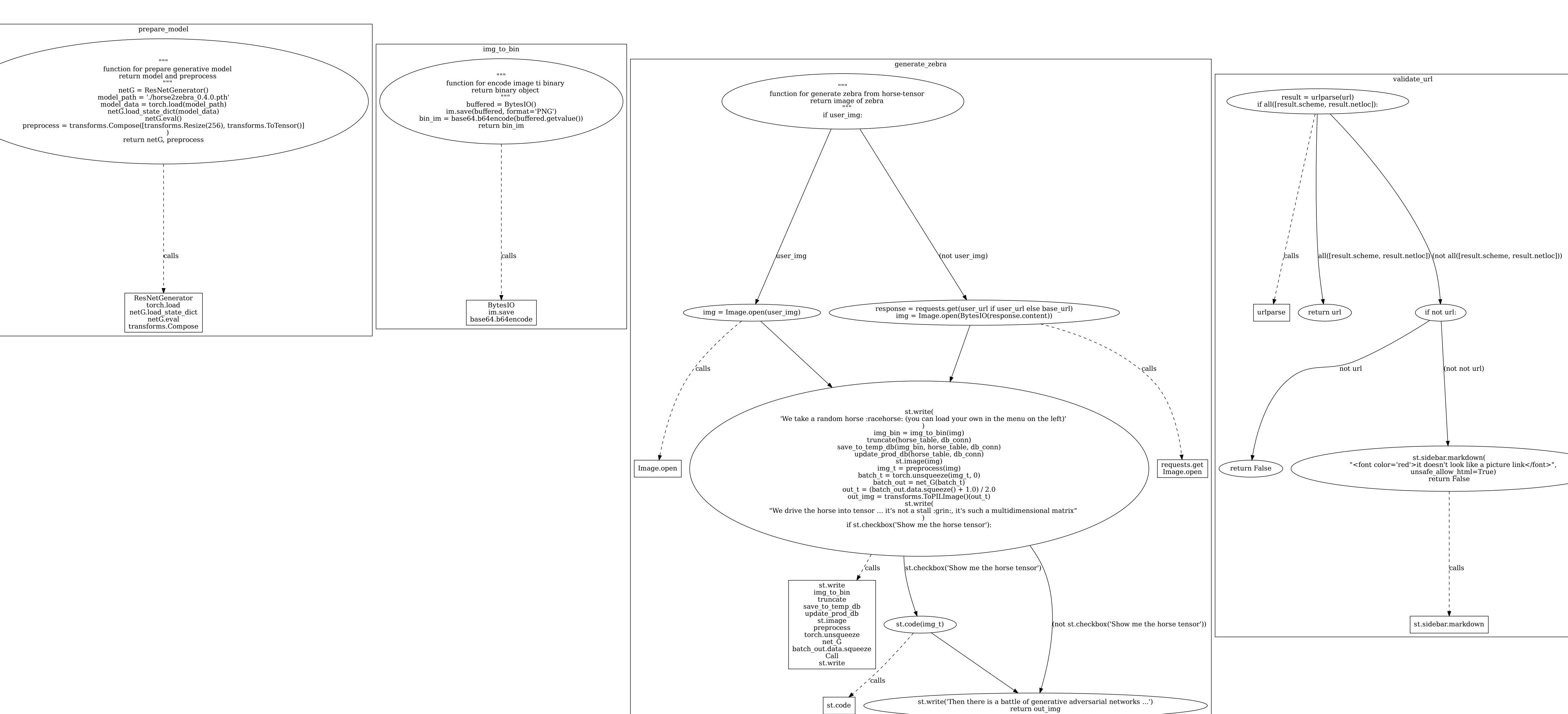
import streamlit as st import requests import torch from torchvision import transforms from resnet import ResNetGenerator
from savedb import truncate, save_to_temp_db, update_prod_db, connect_to_db from urllib.parse import urlparse from PIL import Image from io import BytesIO import base64 st.set_option('deprecation.showfileUploaderEncoding', False) st.title('Zebrate app') st.header('neural network that turns a horse into a zebra') FILE_TYPES = ['png', 'jpg', 'jpeg'] 'https://external-content.duckduckgo.com/iu/?u=https%3A%2F%2Flifeglobe.net%2Fmedia%2Fentry%2F6445%2F1a.jpg' horse_table = 'horse_files'
zebra_table = 'zebra_files'
def uploader(file):...
def prepare_model():...
def img_to_bin(im):... def generate_zebra(net_G, preprocess, user_img, user_url, base_url, db_conn):...

def validate_url(url):...

def main():... if __name__ == '__main__': __name__ == '__main__'





* [Streamlit](https://www.streamlit.io/)

* [Book](https://pytorch.org/assets/deep-learning/Deep-Learning-with-PyTorch.pdf)

* [PyTorch](https://pytorch.org/)

* [CycleGAN](https://github.com/keras-team/keras-io/blob/master/examples/generative/cyclegan.py)

* [ResNet](https://www.res.net/)

* [weights for model](https://github.com/deep-learning-with-pytorch/dlwpt-code/blob/master/data/p1ch2/horse2zebra_0.4.0.pth)

* [and this dataset](http://mng.bz/8pKP)

by [Dmitry Kosarevsky](https://github.com/dKosarevsky) for [TaDS labs](https://networking-labs.ru/) in [BMSTU](https://bmstu.ru)

validate_url
prepare_model
uploader
connect_to_db
generate_zebra
img_to_bin
truncate

save_to_temp_db update_prod_db st.write st.image st.sidebar.markdown

orate.pv