```
from flask import Blueprint
from backend.auth_middleware import token_required
import ibm_boto3
from ibm_botocore.client import Config, ClientError
from backend import config

cos = ibm_boto3.resource("s3",
                         ibm_api_key_id=config["COS_API_KEY_ID"],
                         ibm_service_instance_id=config["COS_INSTANCE_CRN"],
                         config=Config(signature_version="oauth"),
                         endpoint_url=config["COS_ENDPOINT"]
                         )

files = Blueprint("files", __name__)



def multi_part_upload(bucket_name, item_name, file_path):
    try:
        print("Starting file transfer for {0} to bucket: {1}\n".format(
            item_name, bucket_name))
        # set 5 MB chunks
        part_size = 1024 * 1024 * 5

        # set threadhold to 15 MB
        file_threshold = 1024 * 1024 * 15

        # set the transfer threshold and chunk size
        transfer_config = ibm_boto3.s3.transfer.TransferConfig(
            multipart_threshold=file_threshold,
            multipart_chunksize=part_size
        )
    # the upload_fileobj method will automatically execute a multi-part upload
    # in 5 MB chunks for all files over 15 MB
    with open(file_path, "rb") as file_data:
      cos.Object(bucket_name, item_name).upload_fileobj(
        Fileobj=file_data,
        Config=transfer_config
      )

    print("Transfer for {0} Complete!\n".format(item_name))
  except ClientError as be:
    print("CLIENT ERROR: {0}\n".format(be))
  except Exception as e:
    print("Unable to complete multi-part upload: {0}".format(e))


@files.route('/avatar', methods=["POST"])
@token_required
def upload_profile_photo(current_user):
  return "hello"
```