Flask Request Open Source Report

General Information and Licensing

Code Repository	• **Flask**: https://github.com/pallets/flask/ • **werkzeug**: https://github.com/pallets/werkzeug
License Type	• **Flask**: BSD-3-Clause Source License • **werkzeug**: BSD-3-Clause License
License Description	Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
License Restrictions	THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Who worked with this?	Sungho, Tariq, Ahsan

Class flask.Request

Purpose

What does this tech do for you in your project?	This class handles receiving and parsing HTTP requests from packets.
Where specifically is this tech used in your project?	Throughout /flask_server.py as part of class Flask instantiation (https://github.com/rdstrach/CSE312_Group_Project/blob/main/flask_server.py)

Magic

Documentation	• https://flask.palletsprojects.com/en/2.1.x/api/#flask.Request • https://werkzeug.palletsprojects.com/en/2.1.x/wrappers/
How does this technology do what it does for you in the Purpose section of this report?	• The class flask.Request is a subclass of the class werkzeug.wrappers.Request that represents an incoming WSGI HTTP request with headers and body parsed from the WSGI environment. • The WSGI environment containing the HTTP headers and information/body from the WSGI is contained in “environ”. • The data contained in “environ” is passed to werkzeug.wrappers.Request, where it then parses that information in werkzeug.wrappers.Request.__init__(). • In werkzeug.Request.__init__(), it makes get() method calls from various classes in datastructures.py and datastructures.pyi such as: 1. werkzeug.TypeconversionDict 2. werkzeug.Headers 3. werkzeug.CombinedMultiDict • These get() methods then retrieve different parts of the http request (ex. method, path, headers) from “environ” using key-value lookups. • The body can then be retrieved from werkzeug.wrappers.Request.get_data() method that eventually calls upon various parsing functions in http.py and formparser.py that first determine what kind of data it is. • All this HTTP request information is stored in the werkzeug.wrappers.Request object, and accessible to the flask.Request object in the same manner.

Where is the specific code that does what you use the tech for?	• flask.Request: https://github.com/pallets/flask/blob/a03719b01076a5bfdc2c8f4024eda7b874614bc1/src/flask/wrappers.py#L15 • werkzeug.wappers.Request: https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/wrappers/request.py#L29 • environ: https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/wrappers/request.py#L92 • werkzeug.wrappers.Request.__init__(): https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/wrappers/request.py#L99 • werkzeug/datastructures.py: https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/datastructures.py • werkzeug/datastructures.pyi: https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/datastructures.pyi • werkzeug.TypeConversionDict: https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/datastructures.py#L230 • werkzeug.Headers: https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/datastructures.py#L848 • wekzerug.CombinedMultiDict: https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/datastructures.py#L1405 • werkzeug.Request.get_data(): https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/wrappers/request.py#L374 • werkzeug/http.py: https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/http.py • werkzeug/formparser.py https://github.com/pallets/werkzeug/blob/c7ae2fea4fb229ffd71187c2b665874c91b96277/src/werkzeug/formparser.py