[17:31:50] WOULD EXECUTE SQL: INSERT INTO tour\_requests (secret\_id, tour\_id, request\_string, status, started\_at) VALUES ('USER-281301397', 'tour\_199a1b0509c', 'Macsherry Library

Alexandria Bay, NY', 'started', '2025-10-01T17:31:50.843869')

[17:32:00] POLL: Status response: {created\_at: 2025-10-01T21:31:50.489245, job\_id: c84e953c-99d6-4f8a-af1f-5e3419aa0617, location: Macsherry Library

Alexandria Bay, NY, progress: Text generation: Starting tour text generation..., status: processing, total\_stops: 3, tour\_type: museum}

[17:32:00] POLL: Status value: processing

[17:32:00] POLL: Progress value: Text generation: Starting tour text generation...

[17:32:10] POLL: Status response: {created\_at: 2025-10-01T21:31:50.489245, job\_id: c84e953c-99d6-4f8a-af1f-5e3419aa0617, location: Macsherry Library

Alexandria Bay, NY, progress: Waiting for modernized processing..., status: processing, total\_stops: 3, tour\_type: museum}

[17:32:10] POLL: Status value: processing

[17:32:10] POLL: Progress value: Waiting for modernized processing...

[17:32:20] POLL: Status response: {created\_at: 2025-10-01T21:31:50.489245, job\_id: c84e953c-99d6-4f8a-af1f-5e3419aa0617, location: Macsherry Library

Alexandria Bay, NY, progress: Modernized processing: Generating audio files..., status: processing, total\_stops: 3, tour\_type: museum}

[17:32:20] POLL: Status value: processing

[17:32:20] POLL: Progress value: Modernized processing: Generating audio files...

[17:32:30] POLL: Status response: {coordinates: [44.3362, -75.9173], created\_at: 2025-10-01T21:31:50.489245, extract\_dir: macsherry\_library

alexandria\_bay\_ny\_museum\_c84e953c, job\_id: c84e953c-99d6-4f8a-af1f-5e3419aa0617, location: Macsherry Library

Alexandria Bay, NY, netlify\_ready: true, output\_zip: macsherry\_library

alexandria\_bay\_ny\_museum\_c84e953c.zip, progress: MODERNIZED tour generation completed! Tour now has separate MP3 and TXT files for editing., status: completed, total\_stops: 3, tour\_type: museum}

[17:32:30] POLL: Status value: completed

[17:32:30] POLL: Progress value: MODERNIZED tour generation completed! Tour now has separate MP3 and TXT files for editing.

[17:32:30] POLL: Detected completion, starting download...

[17:32:30] POLL: Calling \_autoDownloadAndPlay...

[17:32:30] DOWNLOAD: Starting download for jobId: c84e953c-99d6-4f8a-af1f-5e3419aa0617, location: Macsherry Library

Alexandria Bay, NY

[17:32:30] DOWNLOAD: Requesting download from: http://192.168.1.20:5002/download/c84e953c-99d6-4f8a-af1f-5e3419aa0617

[17:32:30] DOWNLOAD: Response status: 500

[17:32:30] DOWNLOAD: Response size: 29035 bytes

[17:32:30] DOWNLOAD: Failed with status 500: <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>ValueError: Detected newline in header value. This is a potential security problem // Werkzeug Debugger</title>

<link rel="stylesheet" href="?\_\_debugger\_\_=yes&amp;cmd=resource&amp;f=style.css"

type="text/css">

<!-- We need to make sure this has a favicon so that the debugger does

not by accident trigger a request to /favicon.ico which might

change the application state. -->

<link rel="shortcut icon"

href="?\_\_debugger\_\_=yes&amp;cmd=resource&amp;f=console.png">

<script src="?\_\_debugger\_\_=yes&amp;cmd=resource&amp;f=jquery.js"></script>

<script src="?\_\_debugger\_\_=yes&amp;cmd=resource&amp;f=debugger.js"></script>

<script type="text/javascript">

var TRACEBACK = 140368695400672,

CONSOLE\_MODE = false,

EVALEX = true,

EVALEX\_TRUSTED = false,

SECRET = "igCjNWYfIfEJ2oNpMb3R";

</script>

</head>

<body style="background-color: #fff">

<div class="debugger">

<h1>ValueError</h1>

<div class="detail">

<p class="errormsg">ValueError: Detected newline in header value. This is a potential security problem</p>

</div>

<h2 class="traceback">Traceback <em>(most recent call last)</em></h2>

<div class="traceback">

<ul><li><div class="frame" id="frame-140368695401440">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/app.py"</cite>,

line <em class="line">2464</em>,

in <code class="function">\_\_call\_\_</code></h4>

<div class="source library"><pre class="line before"><span class="ws"></span> </pre>

<pre class="line before"><span class="ws"> </span>def \_\_call\_\_(self, environ, start\_response):</pre>

<pre class="line before"><span class="ws"> </span>&quot;&quot;&quot;The WSGI server calls the Flask application object as the</pre>

<pre class="line before"><span class="ws"> </span>WSGI application. This calls :meth:`wsgi\_app` which can be</pre>

<pre class="line before"><span class="ws"> </span>wrapped to applying middleware.&quot;&quot;&quot;</pre>

<pre class="line current"><span class="ws"> </span>return self.wsgi\_app(environ, start\_response)</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>def \_\_repr\_\_(self):</pre>

<pre class="line after"><span class="ws"> </span>return &quot;&lt;%s %r&gt;&quot; % (self.\_\_class\_\_.\_\_name\_\_, self.name)</pre></div>

</div>

<li><div class="frame" id="frame-140368695401824">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/app.py"</cite>,

line <em class="line">2450</em>,

in <code class="function">wsgi\_app</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span>try:</pre>

<pre class="line before"><span class="ws"> </span>ctx.push()</pre>

<pre class="line before"><span class="ws"> </span>response = self.full\_dispatch\_request()</pre>

<pre class="line before"><span class="ws"> </span>except Exception as e:</pre>

<pre class="line before"><span class="ws"> </span>error = e</pre>

<pre class="line current"><span class="ws"> </span>response = self.handle\_exception(e)</pre>

<pre class="line after"><span class="ws"> </span>except: # noqa: B001</pre>

<pre class="line after"><span class="ws"> </span>error = sys.exc\_info()[1]</pre>

<pre class="line after"><span class="ws"> </span>raise</pre>

<pre class="line after"><span class="ws"> </span>return response(environ, start\_response)</pre>

<pre class="line after"><span class="ws"> </span>finally:</pre></div>

</div>

<li><div class="frame" id="frame-140368695401008">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask\_cors/extension.py"</cite>,

line <em class="line">165</em>,

in <code class="function">wrapped\_function</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span># Wrap exception handlers with cross\_origin</pre>

<pre class="line before"><span class="ws"> </span># These error handlers will still respect the behavior of the route</pre>

<pre class="line before"><span class="ws"> </span>if options.get('intercept\_exceptions', True):</pre>

<pre class="line before"><span class="ws"> </span>def \_after\_request\_decorator(f):</pre>

<pre class="line before"><span class="ws"> </span>def wrapped\_function(\*args, \*\*kwargs):</pre>

<pre class="line current"><span class="ws"> </span>return cors\_after\_request(app.make\_response(f(\*args, \*\*kwargs)))</pre>

<pre class="line after"><span class="ws"> </span>return wrapped\_function</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>if hasattr(app, 'handle\_exception'):</pre>

<pre class="line after"><span class="ws"> </span>app.handle\_exception = \_after\_request\_decorator(</pre>

<pre class="line after"><span class="ws"> </span>app.handle\_exception)</pre></div>

</div>

<li><div class="frame" id="frame-140368695398560">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/app.py"</cite>,

line <em class="line">1867</em>,

in <code class="function">handle\_exception</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span># if we want to repropagate the exception, we can attempt to</pre>

<pre class="line before"><span class="ws"> </span># raise it with the whole traceback in case we can do that</pre>

<pre class="line before"><span class="ws"> </span># (the function was actually called from the except part)</pre>

<pre class="line before"><span class="ws"> </span># otherwise, we just raise the error again</pre>

<pre class="line before"><span class="ws"> </span>if exc\_value is e:</pre>

<pre class="line current"><span class="ws"> </span>reraise(exc\_type, exc\_value, tb)</pre>

<pre class="line after"><span class="ws"> </span>else:</pre>

<pre class="line after"><span class="ws"> </span>raise e</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>self.log\_exception((exc\_type, exc\_value, tb))</pre>

<pre class="line after"><span class="ws"> </span>server\_error = InternalServerError()</pre></div>

</div>

<li><div class="frame" id="frame-140368695398848">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/\_compat.py"</cite>,

line <em class="line">39</em>,

in <code class="function">reraise</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span>import collections.abc as collections\_abc</pre>

<pre class="line before"><span class="ws"></span> </pre>

<pre class="line before"><span class="ws"> </span>def reraise(tp, value, tb=None):</pre>

<pre class="line before"><span class="ws"> </span>if value.\_\_traceback\_\_ is not tb:</pre>

<pre class="line before"><span class="ws"> </span>raise value.with\_traceback(tb)</pre>

<pre class="line current"><span class="ws"> </span>raise value</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>implements\_to\_string = \_identity</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"></span>else:</pre>

<pre class="line after"><span class="ws"> </span>iterkeys = lambda d: d.iterkeys()</pre></div>

</div>

<li><div class="frame" id="frame-140368695399280">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/app.py"</cite>,

line <em class="line">2447</em>,

in <code class="function">wsgi\_app</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span>ctx = self.request\_context(environ)</pre>

<pre class="line before"><span class="ws"> </span>error = None</pre>

<pre class="line before"><span class="ws"> </span>try:</pre>

<pre class="line before"><span class="ws"> </span>try:</pre>

<pre class="line before"><span class="ws"> </span>ctx.push()</pre>

<pre class="line current"><span class="ws"> </span>response = self.full\_dispatch\_request()</pre>

<pre class="line after"><span class="ws"> </span>except Exception as e:</pre>

<pre class="line after"><span class="ws"> </span>error = e</pre>

<pre class="line after"><span class="ws"> </span>response = self.handle\_exception(e)</pre>

<pre class="line after"><span class="ws"> </span>except: # noqa: B001</pre>

<pre class="line after"><span class="ws"> </span>error = sys.exc\_info()[1]</pre></div>

</div>

<li><div class="frame" id="frame-140368695399328">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/app.py"</cite>,

line <em class="line">1952</em>,

in <code class="function">full\_dispatch\_request</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span>request\_started.send(self)</pre>

<pre class="line before"><span class="ws"> </span>rv = self.preprocess\_request()</pre>

<pre class="line before"><span class="ws"> </span>if rv is None:</pre>

<pre class="line before"><span class="ws"> </span>rv = self.dispatch\_request()</pre>

<pre class="line before"><span class="ws"> </span>except Exception as e:</pre>

<pre class="line current"><span class="ws"> </span>rv = self.handle\_user\_exception(e)</pre>

<pre class="line after"><span class="ws"> </span>return self.finalize\_request(rv)</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>def finalize\_request(self, rv, from\_error\_handler=False):</pre>

<pre class="line after"><span class="ws"> </span>&quot;&quot;&quot;Given the return value from a view function this finalizes</pre>

<pre class="line after"><span class="ws"> </span>the request by converting it into a response and invoking the</pre></div>

</div>

<li><div class="frame" id="frame-140368695400864">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask\_cors/extension.py"</cite>,

line <em class="line">165</em>,

in <code class="function">wrapped\_function</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span># Wrap exception handlers with cross\_origin</pre>

<pre class="line before"><span class="ws"> </span># These error handlers will still respect the behavior of the route</pre>

<pre class="line before"><span class="ws"> </span>if options.get('intercept\_exceptions', True):</pre>

<pre class="line before"><span class="ws"> </span>def \_after\_request\_decorator(f):</pre>

<pre class="line before"><span class="ws"> </span>def wrapped\_function(\*args, \*\*kwargs):</pre>

<pre class="line current"><span class="ws"> </span>return cors\_after\_request(app.make\_response(f(\*args, \*\*kwargs)))</pre>

<pre class="line after"><span class="ws"> </span>return wrapped\_function</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>if hasattr(app, 'handle\_exception'):</pre>

<pre class="line after"><span class="ws"> </span>app.handle\_exception = \_after\_request\_decorator(</pre>

<pre class="line after"><span class="ws"> </span>app.handle\_exception)</pre></div>

</div>

<li><div class="frame" id="frame-140368695398608">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/app.py"</cite>,

line <em class="line">1821</em>,

in <code class="function">handle\_user\_exception</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span>return self.handle\_http\_exception(e)</pre>

<pre class="line before"><span class="ws"></span> </pre>

<pre class="line before"><span class="ws"> </span>handler = self.\_find\_error\_handler(e)</pre>

<pre class="line before"><span class="ws"></span> </pre>

<pre class="line before"><span class="ws"> </span>if handler is None:</pre>

<pre class="line current"><span class="ws"> </span>reraise(exc\_type, exc\_value, tb)</pre>

<pre class="line after"><span class="ws"> </span>return handler(e)</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>def handle\_exception(self, e):</pre>

<pre class="line after"><span class="ws"> </span>&quot;&quot;&quot;Handle an exception that did not have an error handler</pre>

<pre class="line after"><span class="ws"> </span>associated with it, or that was raised from an error handler.</pre></div>

</div>

<li><div class="frame" id="frame-140368669965856">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/\_compat.py"</cite>,

line <em class="line">39</em>,

in <code class="function">reraise</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span>import collections.abc as collections\_abc</pre>

<pre class="line before"><span class="ws"></span> </pre>

<pre class="line before"><span class="ws"> </span>def reraise(tp, value, tb=None):</pre>

<pre class="line before"><span class="ws"> </span>if value.\_\_traceback\_\_ is not tb:</pre>

<pre class="line before"><span class="ws"> </span>raise value.with\_traceback(tb)</pre>

<pre class="line current"><span class="ws"> </span>raise value</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>implements\_to\_string = \_identity</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"></span>else:</pre>

<pre class="line after"><span class="ws"> </span>iterkeys = lambda d: d.iterkeys()</pre></div>

</div>

<li><div class="frame" id="frame-140368669966048">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/app.py"</cite>,

line <em class="line">1950</em>,

in <code class="function">full\_dispatch\_request</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span>self.try\_trigger\_before\_first\_request\_functions()</pre>

<pre class="line before"><span class="ws"> </span>try:</pre>

<pre class="line before"><span class="ws"> </span>request\_started.send(self)</pre>

<pre class="line before"><span class="ws"> </span>rv = self.preprocess\_request()</pre>

<pre class="line before"><span class="ws"> </span>if rv is None:</pre>

<pre class="line current"><span class="ws"> </span>rv = self.dispatch\_request()</pre>

<pre class="line after"><span class="ws"> </span>except Exception as e:</pre>

<pre class="line after"><span class="ws"> </span>rv = self.handle\_user\_exception(e)</pre>

<pre class="line after"><span class="ws"> </span>return self.finalize\_request(rv)</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>def finalize\_request(self, rv, from\_error\_handler=False):</pre></div>

</div>

<li><div class="frame" id="frame-140368669963504">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/app.py"</cite>,

line <em class="line">1936</em>,

in <code class="function">dispatch\_request</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span>getattr(rule, &quot;provide\_automatic\_options&quot;, False)</pre>

<pre class="line before"><span class="ws"> </span>and req.method == &quot;OPTIONS&quot;</pre>

<pre class="line before"><span class="ws"> </span>):</pre>

<pre class="line before"><span class="ws"> </span>return self.make\_default\_options\_response()</pre>

<pre class="line before"><span class="ws"> </span># otherwise dispatch to the handler for that endpoint</pre>

<pre class="line current"><span class="ws"> </span>return self.view\_functions[rule.endpoint](\*\*req.view\_args)</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>def full\_dispatch\_request(self):</pre>

<pre class="line after"><span class="ws"> </span>&quot;&quot;&quot;Dispatches the request and on top of that performs request</pre>

<pre class="line after"><span class="ws"> </span>pre and postprocessing as well as HTTP exception catching and</pre>

<pre class="line after"><span class="ws"> </span>error handling.</pre></div>

</div>

<li><div class="frame" id="frame-140368669963408">

<h4>File <cite class="filename">"/app/tour\_orchestrator\_service.py"</cite>,

line <em class="line">735</em>,

in <code class="function">download\_tour</code></h4>

<div class="source "><pre class="line before"><span class="ws"> </span>if not os.path.exists(zip\_path):</pre>

<pre class="line before"><span class="ws"> </span>print(f&quot;File not found: {zip\_path}&quot;)</pre>

<pre class="line before"><span class="ws"> </span>return jsonify({&quot;error&quot;: &quot;File not found&quot;}), 404</pre>

<pre class="line before"><span class="ws"></span> </pre>

<pre class="line before"><span class="ws"> </span>print(f&quot;Sending file: {zip\_path}&quot;)</pre>

<pre class="line current"><span class="ws"> </span>return send\_file(zip\_path, as\_attachment=True, attachment\_filename=job[&quot;output\_zip&quot;])</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"></span>@app.route('/serve/&lt;job\_id&gt;', methods=['GET'])</pre>

<pre class="line after"><span class="ws"></span>def serve\_tour\_info(job\_id):</pre>

<pre class="line after"><span class="ws"> </span>&quot;&quot;&quot;Get information about serving the tour.&quot;&quot;&quot;</pre>

<pre class="line after"><span class="ws"> </span>print(f&quot;\n==== SERVE REQUEST: {datetime.now().isoformat()} ====&quot;)</pre></div>

</div>

<li><div class="frame" id="frame-140368669966192">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/flask/helpers.py"</cite>,

line <em class="line">618</em>,

in <code class="function">send\_file</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span>&quot;filename\*&quot;: &quot;UTF-8''%s&quot; % url\_quote(attachment\_filename, safe=b&quot;&quot;),</pre>

<pre class="line before"><span class="ws"> </span>}</pre>

<pre class="line before"><span class="ws"> </span>else:</pre>

<pre class="line before"><span class="ws"> </span>filenames = {&quot;filename&quot;: attachment\_filename}</pre>

<pre class="line before"><span class="ws"></span> </pre>

<pre class="line current"><span class="ws"> </span>headers.add(&quot;Content-Disposition&quot;, &quot;attachment&quot;, \*\*filenames)</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>if current\_app.use\_x\_sendfile and filename:</pre>

<pre class="line after"><span class="ws"> </span>if file is not None:</pre>

<pre class="line after"><span class="ws"> </span>file.close()</pre>

<pre class="line after"><span class="ws"> </span>headers[&quot;X-Sendfile&quot;] = filename</pre></div>

</div>

<li><div class="frame" id="frame-140368669964656">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/werkzeug/datastructures.py"</cite>,

line <em class="line">1188</em>,

in <code class="function">add</code></h4>

<div class="source library"><pre class="line before"><span class="ws"> </span>&quot;&quot;&quot;</pre>

<pre class="line before"><span class="ws"> </span>if kw:</pre>

<pre class="line before"><span class="ws"> </span>\_value = \_options\_header\_vkw(\_value, kw)</pre>

<pre class="line before"><span class="ws"> </span>\_key = \_unicodify\_header\_value(\_key)</pre>

<pre class="line before"><span class="ws"> </span>\_value = \_unicodify\_header\_value(\_value)</pre>

<pre class="line current"><span class="ws"> </span>self.\_validate\_value(\_value)</pre>

<pre class="line after"><span class="ws"> </span>self.\_list.append((\_key, \_value))</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>def \_validate\_value(self, value):</pre>

<pre class="line after"><span class="ws"> </span>if not isinstance(value, text\_type):</pre>

<pre class="line after"><span class="ws"> </span>raise TypeError(&quot;Value should be unicode.&quot;)</pre></div>

</div>

<li><div class="frame" id="frame-140368669962784">

<h4>File <cite class="filename">"/usr/local/lib/python3.9/site-packages/werkzeug/datastructures.py"</cite>,

line <em class="line">1195</em>,

in <code class="function">\_validate\_value</code></h4>

<div class="source library"><pre class="line before"><span class="ws"></span> </pre>

<pre class="line before"><span class="ws"> </span>def \_validate\_value(self, value):</pre>

<pre class="line before"><span class="ws"> </span>if not isinstance(value, text\_type):</pre>

<pre class="line before"><span class="ws"> </span>raise TypeError(&quot;Value should be unicode.&quot;)</pre>

<pre class="line before"><span class="ws"> </span>if u&quot;\n&quot; in value or u&quot;\r&quot; in value:</pre>

<pre class="line current"><span class="ws"> </span>raise ValueError(</pre>

<pre class="line after"><span class="ws"> </span>&quot;Detected newline in header value. This is &quot;</pre>

<pre class="line after"><span class="ws"> </span>&quot;a potential security problem&quot;</pre>

<pre class="line after"><span class="ws"> </span>)</pre>

<pre class="line after"><span class="ws"></span> </pre>

<pre class="line after"><span class="ws"> </span>def add\_header(self, \_key, \_value, \*\*\_kw):</pre></div>

</div>

</ul>

<blockquote>ValueError: Detected newline in header value. This is a potential security problem</blockquote>

</div>

<div class="plain">

<form action="/?\_\_debugger\_\_=yes&amp;cmd=paste" method="post">

<p>

<input type="hidden" name="language" value="pytb">

This is the Copy/Paste friendly version of the traceback. <span

class="pastemessage">You can also paste this traceback into

a <a href="https://gist.github.com/">gist</a>:

<input type="submit" value="create paste"></span>

</p>

<textarea cols="50" rows="10" name="code" readonly>Traceback (most recent call last):

File &quot;/usr/local/lib/python3.9/site-packages/flask/app.py&quot;, line 2464, in \_\_call\_\_

return self.wsgi\_app(environ, start\_response)

File &quot;/usr/local/lib/python3.9/site-packages/flask/app.py&quot;, line 2450, in wsgi\_app

response = self.handle\_exception(e)

File &quot;/usr/local/lib/python3.9/site-packages/flask\_cors/extension.py&quot;, line 165, in wrapped\_function

return cors\_after\_request(app.make\_response(f(\*args, \*\*kwargs)))

File &quot;/usr/local/lib/python3.9/site-packages/flask/app.py&quot;, line 1867, in handle\_exception

reraise(exc\_type, exc\_value, tb)

File &quot;/usr/local/lib/python3.9/site-packages/flask/\_compat.py&quot;, line 39, in reraise

raise value

File &quot;/usr/local/lib/python3.9/site-packages/flask/app.py&quot;, line 2447, in wsgi\_app

response = self.full\_dispatch\_request()

File &quot;/usr/local/lib/python3.9/site-packages/flask/app.py&quot;, line 1952, in full\_dispatch\_request

rv = self.handle\_user\_exception(e)

File &quot;/usr/local/lib/python3.9/site-packages/flask\_cors/extension.py&quot;, line 165, in wrapped\_function

return cors\_after\_request(app.make\_response(f(\*args, \*\*kwargs)))

File &quot;/usr/local/lib/python3.9/site-packages/flask/app.py&quot;, line 1821, in handle\_user\_exception

reraise(exc\_type, exc\_value, tb)

File &quot;/usr/local/lib/python3.9/site-packages/flask/\_compat.py&quot;, line 39, in reraise

raise value

File &quot;/usr/local/lib/python3.9/site-packages/flask/app.py&quot;, line 1950, in full\_dispatch\_request

rv = self.dispatch\_request()

File &quot;/usr/local/lib/python3.9/site-packages/flask/app.py&quot;, line 1936, in dispatch\_request

return self.view\_functions[rule.endpoint](\*\*req.view\_args)

File &quot;/app/tour\_orchestrator\_service.py&quot;, line 735, in download\_tour

return send\_file(zip\_path, as\_attachment=True, attachment\_filename=job[&quot;output\_zip&quot;])

File &quot;/usr/local/lib/python3.9/site-packages/flask/helpers.py&quot;, line 618, in send\_file

headers.add(&quot;Content-Disposition&quot;, &quot;attachment&quot;, \*\*filenames)

File &quot;/usr/local/lib/python3.9/site-packages/werkzeug/datastructures.py&quot;, line 1188, in add

self.\_validate\_value(\_value)

File &quot;/usr/local/lib/python3.9/site-packages/werkzeug/datastructures.py&quot;, line 1195, in \_validate\_value

raise ValueError(

ValueError: Detected newline in header value. This is a potential security problem</textarea>

</form>

</div>

<div class="explanation">

The debugger caught an exception in your WSGI application. You can now

look at the traceback which led to the error. <span class="nojavascript">

If you enable JavaScript you can also use additional features such as code

execution (if the evalex feature is enabled), automatic pasting of the

exceptions and much more.</span>

</div>

<div class="footer">

Brought to you by <strong class="arthur">DON'T PANIC</strong>, your

friendly Werkzeug powered traceback interpreter.

</div>

</div>

<div class="pin-prompt">

<div class="inner">

<h3>Console Locked</h3>

<p>

The console is locked and needs to be unlocked by entering the PIN.

You can find the PIN printed out on the standard output of your

shell that runs the server.

<form>

<p>PIN:

<input type=text name=pin size=14>

<input type=submit name=btn value="Confirm Pin">

</form>

</div>

</div>

</body>

</html>

<!--

Traceback (most recent call last):

File "/usr/local/lib/python3.9/site-packages/flask/app.py", line 2464, in \_\_call\_\_

return self.wsgi\_app(environ, start\_response)

File "/usr/local/lib/python3.9/site-packages/flask/app.py", line 2450, in wsgi\_app

response = self.handle\_exception(e)

File "/usr/local/lib/python3.9/site-packages/flask\_cors/extension.py", line 165, in wrapped\_function

return cors\_after\_request(app.make\_response(f(\*args, \*\*kwargs)))

File "/usr/local/lib/python3.9/site-packages/flask/app.py", line 1867, in handle\_exception

reraise(exc\_type, exc\_value, tb)

File "/usr/local/lib/python3.9/site-packages/flask/\_compat.py", line 39, in reraise

raise value

File "/usr/local/lib/python3.9/site-packages/flask/app.py", line 2447, in wsgi\_app

response = self.full\_dispatch\_request()

File "/usr/local/lib/python3.9/site-packages/flask/app.py", line 1952, in full\_dispatch\_request

rv = self.handle\_user\_exception(e)

File "/usr/local/lib/python3.9/site-packages/flask\_cors/extension.py", line 165, in wrapped\_function

return cors\_after\_request(app.make\_response(f(\*args, \*\*kwargs)))

File "/usr/local/lib/python3.9/site-packages/flask/app.py", line 1821, in handle\_user\_exception

reraise(exc\_type, exc\_value, tb)

File "/usr/local/lib/python3.9/site-packages/flask/\_compat.py", line 39, in reraise

raise value

File "/usr/local/lib/python3.9/site-packages/flask/app.py", line 1950, in full\_dispatch\_request

rv = self.dispatch\_request()

File "/usr/local/lib/python3.9/site-packages/flask/app.py", line 1936, in dispatch\_request

return self.view\_functions[rule.endpoint](\*\*req.view\_args)

File "/app/tour\_orchestrator\_service.py", line 735, in download\_tour

return send\_file(zip\_path, as\_attachment=True, attachment\_filename=job["output\_zip"])

File "/usr/local/lib/python3.9/site-packages/flask/helpers.py", line 618, in send\_file

headers.add("Content-Disposition", "attachment", \*\*filenames)

File "/usr/local/lib/python3.9/site-packages/werkzeug/datastructures.py", line 1188, in add

self.\_validate\_value(\_value)

File "/usr/local/lib/python3.9/site-packages/werkzeug/datastructures.py", line 1195, in \_validate\_value

raise ValueError(

ValueError: Detected newline in header value. This is a potential security problem

-->

[17:32:30] DOWNLOAD: Error occurred: Exception: Failed to download tour

[17:32:31] POLL: \_autoDownloadAndPlay completed

[17:32:31] Looking for tour\_id for job: c84e953c-99d6-4f8a-af1f-5e3419aa0617, found: tour\_199a1b0509c

[17:32:31] Updating status for tour\_id: tour\_199a1b0509c to completed using direct SQL

[17:32:31] DIRECT UPDATE: Updating tour\_id=tour\_199a1b0509c to status=completed

[17:32:31] DIRECT UPDATE RESPONSE: 200 - {

"rows\_affected": 1,

"status": "success",

"tour\_id": "tour\_199a1b0509c"

}

[17:32:31] Successfully updated tour status via direct update API: c84e953c-99d6-4f8a-af1f-5e3419aa0617 to completed with tour\_id: tour\_199a1b0509c

[17:34:37] Checking user: USER-281301397

[17:34:37] Testing connectivity to: 192.168.1.20

[17:34:37] ✅ Server connectivity: OK