MSPaintr: gevent-websocket Report

Github:https://gitlab.com/noppo/gevent-websocket/-/blob/master/geventwebsocket/websocket.py

1. What does this technology (library/framework/service) accomplish for you?

gevent-websockets is a websocket library which helps us update our news feed (posts) and posts (upvotes and comments) without the need for a page refresh.

Send

We will use this method to send frames containing the bytes for images uploaded in new posts for the news feed. Upvoting and commenting on a post also utilizes this method to send information to the server.

Receive

The receive method allows us to retrieve newly updated posts on the news feed. As for a post, it will let our site show an updated comment thread and number of upvotes in “real time”.

2. How does this technology accomplish what it does?

Establishing the websocket connection:

> **GUID**
>
> https://github.com/jgelens/gevent-websocket/blob/master/geventwebsocket/handler.py#L32
>
> https://github.com/jgelens/gevent-websocket/blob/master/geventwebsocket/handler.py#L229
>
> **Upgrading the connection**
> https://github.com/jgelens/gevent-websocket/blob/master/geventwebsocket/handler.py#L139
>
> https://github.com/jgelens/gevent-websocket/blob/master/geventwebsocket/handler.py#L244

The WebSocketHandler class upgrades the current connection to a websocket connection. run_application() (Line 65) calls upgrade_websocket() (Line 92), which then calls upgrade_connection(). Upgrade_connection calls start_response() with “101 Switching Protocols” on line 244. These functions perform the process for the websocket handshake (we have discussed the protocol in lecture, so it would be redundant to list the steps out again).

Send:

https://gitlab.com/noppo/gevent-websocket/-/blob/master/geventwebsocket/websocket.py#L366

Send first checks whether the bytes to be sent are that of a binary frame. If so, set the opcode to OPCODE_BINARY in the frame. Otherwise, set it to OPCODE_TEXT to indicate the frame will be a text frame. Afterwards, try to call send_frame() (Line 334). Upon an error from send_frame(), immediately close the websocket connection and throw an WebSocketError error.

Send_frame() first checks that the connection is not closed, and will raise a WebSocketError if it is. Additionally, it checks that message is not null (it simply returns if it is). Next, depending on the opcode, it will either call encode_bytes (for a text frame) or bytes (for a binary frame) on the message to be sent. Send_frame also has the option to compress the message, but is not relevant to our project. After the message is encoded, send_frame() calls encode_header() to set the header values. Finally send_frame() attempts to write the header and message through raw_write().

Receive:

https://gitlab.com/noppo/gevent-websocket/-/blob/master/geventwebsocket/websocket.py#L309

Receive first checks that the websocket connection is closed. If so, raise a WebSocketError. Otherwise, try to read the message with read_message() (Line 249). If read_message returns an error (protocol error, timeout, etc.), close the websocket connection immediately.

Read_message() first extracts the header and payload values with read_frame() (Line 193). Then, while there are frames to be read, get the opcode from the header value. If the opcode matches that of a text or binary frame and if the frame is the first frame received, keep track of the opcode value in another variable to be used for comparison with subsequent frames. If the opcode in subsequent frames does not match, raise a

ProtocolError error. Finally, if no other error occurs, extract the payload from the frame and return it back to receive().

3. What license(s) or terms of service apply to this technology?

https://gitlab.com/noppo/gevent-websocket/-/blob/master/LICENSE

Gevent-websocket is under the Apache License and is copyrighted by Jeffrey Gelens, 2011-2017.

The Apache License states that software under the license may be freely used, reproduced, modified, distributed and sold. Software derived from existing license software does not have to be distributed under the same license. However, such software may not be redistributed without any attribution.

Gevent-websocket is also given “AS IS”, meaning it does not have any express or implied warranties. The creators and copyright holders are not liable for any damages for those who wish to use the software. For our project, this means that we are free to use gevent-websocket with no charge. However, if anything happens to our project as a result of using this library, the creators cannot be held responsible.