3/31:

Completed:

- Read the HTTP/3 specification to identify the requirements to track and test for
    - Prioritized ensuring the correct errors were raised
    - Identified several errors to test for
    - **H3_NO_ERROR** (0x0100):
        - No error. This is used when the connection or stream needs to be closed, but there is no error to signal.
    - **H3_FRAME_ERROR** (0x0106):
        - A frame that fails to satisfy layout requirements or with an invalid size was received.
    - **H3_SETTINGS_ERROR** (0x0109):
        - An endpoint detected an error in the payload of a SETTINGS frame.
    - **H3_REQUEST_INCOMPLETE** (0x010d):
        - The client's stream terminated without containing a fully-formed request.
    - **H3_MESSAGE_ERROR** (0x010e):
        - An HTTP message was malformed and cannot be processed.
- Identified a set of HTTP/3 implementations to test against
    - Implementations within a language / framework
        - Node.js https://nodejs.org/api/http.html
        - **Java** HTTP Server class https://docs.oracle.com/javase/8/docs/jre/api/net/httpserver/spec/com/sun/net/httpserver/HttpServer.html
        - Python HTTP server module https://docs.python.org/3/library/http.server.html
    - Standalone
        - **Apache HTTP server** https://httpd.apache.org/
        - Nginx https://www.nginx.com/
        - Lightpd https://www.lighttpd.net/

TODO:

- Install Java HTTP Server and Apache (locally, with a simple Hello World app on top)
    - Write test cases of normal behavior
    - Possibly try to trigger H3_FRAME_ERROR
    - Log outputs and look for correctness manually
- Program one set of test cases to run against an implementation