We want to check:

1) The validity of all project-related HTML pages in the project directory space as HTML5 with XML syntax. We can’t do this with XML-based validation because XML-based validation can’t check for @data- attributes, so I think we’ll want to get a list of all of the pages and validate them using the W3C non-DTD-based “Nu Markup Checker” validator.  There are instructions in the “HTTP client (for fast command-line checking)” of <http://validator.github.io/validator/#usage> for validating against the Nu Markup validator hosted on the W3C site. I think that’s better than installing our own local copy, primarily because theirs is under constant development, and we don’t want to have to keep updating ours as theirs changes.

For course-specific reasons, if the validator doesn’t do this for us, we need to check for the XML declaration, the correct doctype declaration (either the bare one or the legacy-compat version), and the namespace. We also need to verify that the filename extension is “xhtml” and not “html”; Obdurodon is configured to set the mime type for the former to XML and for the latter to non-XML HTML, and we want the XML value.

2) The validity of all project-related CSS pages in the project directory space. The W3C Jigsaw CSS validator provides a SOAP API (not REST, unfortunately); see <http://jigsaw.w3.org/css-validator/api.html> for documentation.

3) We need to ensure that there are no broken links in pages hosted in project directory space. The W3C has a link-validation tool that I’ve installed on Obdurodon (<http://validator.w3.org/docs/checklink>). Run it by typing “checklink” on the command line. The output is verbose, but perhaps it’s useful for our purposes. We don’t want to validate links that originate outside project space, since the project teams have no control over those anyway.

I think a command-line script that gives simple plain-text output would be best. You might have it default to the current working directory, with the idea that the students would run it from their main project directories, but it would be better if it also took a directory path as an argument, so that instructors could validate multiple projects without having to change directory first.

I don’t think there are any security concerns, since it will run without special privileges and it will be accessing only files that are world-readable.

We don’t teach PHP in this course, so it isn’t much of a concern, but some students do use it. Validating PHP means letting the PHP interpreter Do Its Thing and then validating the resulting HTML. I think the W3C Nu Markup Checker validator should be able to take care of those, but it means that we’ll have to make sure that our wrapper scripts sends it files with the “php” extension, and not just “xhtml”.