DECO3801 Test Plan Document

THEM - Typed HTML5 Evaluation Machine

Carl Hattenfels, Scott Heiner, Shen Yong Lau, Robert Meyer, Brendan Miller, David Uebergang

Functional Test Plan

Testing Strategy

There are three major testable components of the Typed HTML5 Evaluation Machine, our web application: the front-end website, back-end parser and database. While it was easy to write Python test cases for the back-end parser, it was more difficult to test our front-end website and database with a suite of computer-run tests. Instead, we wrote up a series of scenarios that we would undertake to ensure that the web application was running correctly and as expected.

Test Case Transcript

Implications of Functional Testing

User Experience Goals

- very surgical, ambient, passive - the tool should give users immediate insight into the issues with their html / websites and the user can then go and fix it - user shouldn't get invested in the system, nor get frustrated by the errors - it is meant to be a program you just "touch", that is, upload your file you want to check, and then go back and fix it, and then come back to this to validate again, in a cyclic process. priorities are on quick and easy use, which is why everything is instantly accessible and requires very few clicks to navigate.

User Testing Plan

- user groups - undergraduate students who have already done DECO1400 - undergraduate students who haven't done DECO1400 but have worked with computers - masters students who have already done DECO7140 - masters students who have not already done DECO7140

- in general, users had no trouble navigating the system - the result of the validation (i.e. highlighted tags) was not well understood by users

Summarise the results of your tests. For each scenario-based test: tabulate your metric results against each task describe or present your users feedback during/after the test Close with a general discussion that: summarises the issues raised identifies areas for improvement and design suggestions outlines any redefinition of functional and user test plans for final prototype (We dont care whether your prototype passes the tests. What is important is that your prototype is sufficiently broad to validate your test plan for the final product.)