

HTC issue #20

System/Load testing software investigation results – By Jeff Hatton

Considerations:

- Our tech stack – testing software needs to be appropriate for our stack choices, such as having load testing that can work with a particular database.
- Ease of use – software must be reasonable to learn given the time constraints of our project
- Availability – looking only at free to use software that doesn't have usage limitations

Results:

System Testing

As far as system testing goes, no feasible results were yielded – system testing was consistently referred to as a process which differed too much depending on the application. We as a team will need to develop our own process according to our needs for a given milestone. One example of a tutorial was:

<https://www.softwaretestinghelp.com/web-application-testing/>

The above link addresses most of our core functionalities.

Functionality (links, field validation, forms, database, cookies)

Usability (HCI)

Interface testing – team testing/validation of integration tests (frontend to server, server to DB especially)

CSS – likely needs to be manual across multiple browsers and devices/screen sizes

OS testing – manual testing within Linux VM, Window, and Mac

Load Testing

As far as load testing goes, we have two needs – one is testing our http requests for the client/server side code, and the second is testing database functionality/integrity.

Recommendations:

Where appropriate, Apache Jmeter Tests are written in Java, so all team members should be able to write/learn the software, and Jmeter supports testing of MongoDB. Jmeter is open source and fully useable to anyone.

Another alternative is Blazemeter. Accessible, and may offer better support for Javascript or Typescript, and it integrates with Jmeter.