Testing A Web Site

• Want to simulate actual use cases / scenarios
• Play a sequence of actions from start to finish
  • Login through logout
  • Look at what results
• Need to generate user actions
• Need to check the results
• Need to have multiple such scenarios (lots)
Generating User Actions

• Can be done without a browser
  • Actions = URL request with proper context
    • Context = cookies, put fields, ...
  • curl is a command-line tool that can do this
  • Lots of work however
  • But you can put together scripts of curl calls to emulate tests

• Want to do it with a browser
  • Or something that acts as a browser
Web Site Testing Tools

- **httpunit**
  - Create test cases for calls to the server
    - Providing input, checking expected output
    - These are using a Java framework
  - Generating test cases automatically
    - By analyzing on the JavaScript code

- **Sikuli**
  - Test cases with visual input and output
    - Why is this difficult?
  - Examples: [https://www.youtube.com/watch?v=pWLa1kxakOs](https://www.youtube.com/watch?v=pWLa1kxakOs)
  - Overview: [https://www.youtube.com/watch?v=01jFl8KrEMY](https://www.youtube.com/watch?v=01jFl8KrEMY)
Browser Automation: Selenium

• Example: https://www.youtube.com/watch?v=gsHyDlyA3dg
• Installation: http://www.seleniumhq.org/download
• Firefox:
• Chrome:
  • https://github.com/SeleniumHQ/selenium/wiki/ChromeDriver
• Example of creating a script
• Example of a written script
Performance Testing

• Importance of performance
  • 100ms makes the difference between success & failure

• What do you want to test
  • How fast the web site performs
  • Speed to undertake common actions
  • How responsive the web site is
  • What happens if ...

• What are the testing circumstances
  • How many users should you have for testing?
Performance Testing

• Most developer tool sets include this for a single page:
  • Can look at browser debugger network page
• gtmetrix: https://gtmetrix.com
Load Testing

- How does the application behave under load
- What types of load should be considered
  - Network performance
    - How does the user’s connection affect your application
    - How does overall network activity affect your application
  - Large numbers of simultaneous users
    - What happens to your server under load
    - What happens to your database under load
  - Large individual requests
    - Complex database queries
  - Heavy load on specific pages
Stress Testing

• What are the limits of your application
• What types of things to consider
  • Maximum load that can be tolerated
  • Maximum input size that can be processed
• Determine what happens when things go wrong
  • Database crash or disconnect
  • Server crash or disconnect
  • Web server crash or disconnect
  • Browser crash or disconnect
  • Network crash (server/browser stay up)
  • Gradual degradation
How To Do Performance Testing

• What do you need to do
  • Recruit hundreds of users
  • Simulate lots of users
    • Doing normal things with the system
    • Doing particular things with the system
  • Simulate failures

• Suppose I wanted to try my app with 1000 users
  • How might you do this?
Jmeter: Performance Testing

- Jmeter is an open-source web performance tester
  - [http://www.youtube.com/watch?v=8NLeq-QxkSw](http://www.youtube.com/watch?v=8NLeq-QxkSw)
- Works with a set of test cases
  - Series of interactions with the back end
  - These can be specified
    - Manually by a set of HTTP requests (URLs with data)
    - By example (gathering information from sample runs)
- Will run many of these simultaneously
  - You get to specify which ones and how many
  - With random delays
  - For as long as you want
Security Testing

• **URL security**
  - Ability to bypass login/security by creating a URL
  - Ability to get private pages by editing URLs
  - Passing in inputs that will make the system misbehave
    - Overly long inputs that can cause buffer overflows

• **Input checking**
  - Are all invalid inputs detected
  - Openness to SQL injection and Cross-browser attacks

• **Are internal files, etc. in the web pages inaccessible**

• **Is SSL used for all appropriate pages**
  - Can it be bypassed?

• **Are all errors, security breach attempts, etc. logged**
Security Testing

• **Tools exist**
  - Websecureify ([http://vimeo.com/41362046](http://vimeo.com/41362046))
  - Netsparker ([https://www.youtube.com/watch?v=bVpv4r1T5Ac](https://www.youtube.com/watch?v=bVpv4r1T5Ac))
    - Download: ([http://www.netsparker.com](http://www.netsparker.com))
  - Wapiti, Websurgery, ...
Design For Testing

• Want to have a web site that is testable
  • Might not be possible to test live site
    • Don’t want it to crash
    • Want to test before installing updates
    • Actions might have real-world effects

• Set up a test site
  • Separate database
    • Add test users
  • Internal code to handle external actions
    • Based on which server is being run
  • Do it on a local machine / separate VM
  • Possibly special URLs to reset the server to a known state
Lab Preparation

• Have an accessible test site for your project
  • Can be production site if that is safe
• Meet to determine who will test what
  • Html/CSS, Usability, Web Site, Performance, Load Testing, Security
  • Develop a testing plan for your project
  • When and what, reusable tests, testing before update, ...
• Download and install the tools you need if any
• Come prepared to set up tests and test your project
Next Time

• Next Time: Testing Lab