CS1320
Creating Modern Web and Mobile Applications

Lecture 32:

Testing I

Testing

- When looking at security and privacy
  - We keep asking “what can go wrong”
  - What happens if a user does <x> when <y>
- You want to do this in general for your application
  - To make sure it will work
  - To make sure it will keep working
  - To eliminate potential problems before users find them

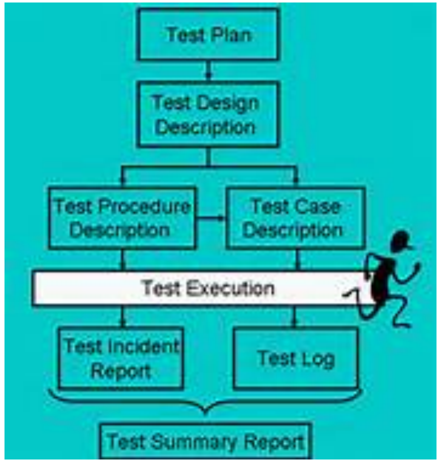
You've Built a Web Application

- What do you really know about it
  - Does it work?
  - Does it work correctly?
  - Does it work correctly under all circumstances?
  - Will users like it? Will they use it?
  - Did you build the right application?
  - Does it have security holes?
  - Will it scale?
- How do you answer such questions?
  - Testing

What is Testing?

- The process of running software in order to find bugs
  - Not to show that bugs are not there
  - What is the difference?
- A successful test case is one that finds a bug
- Good testers (QA) are people who
  - Can sit in front of software and break it
  - Are in the frame of mind where you want to break things
  - Are TAs grading homework assignments
  - Its difficult to test your own software. Much easier to test others
- Testing won't show what's right, just what isn't wrong

Software Testing



- Introduced in 15/16/17/18/32
  - Agile programming: write the test cases first
  - Incremental development: continuous testing
- You've possibly seen tools to help with testing
  - Junit for java testing
    - Test cases are methods annotated with @Test
    - Automatically find and run all tests for a system
  - Supports repeated testing

Regression Testing

- Testing software once is not very useful
  - You might find and fix a problem this way
  - But what if the software changes
- Regression tests
  - Tests that are run each time the system changes
  - Rerun after each change to ensure no regression
- Test cases should be permanent, not throw-away
  - How to do this for web and mobile applications?
  - User interfaces and functionality keep changing

Testing Web & Mobile Applications

- You should test your applications
  - Lots of tools and techniques exist
    - Both commercial and open source
  - Difficulty in testing is no excuse for not doing it
- Testing should be done at all levels
- Testing should be considered from the start
  - Plan a test database, test users, test data, … to facilitate
  - Have a separate server running for test purposes
  - Design the application to facilitate testing
    - For example, allow dummy payments, ordering, …
- We've been doing this throughout the course

Testing Your Projects

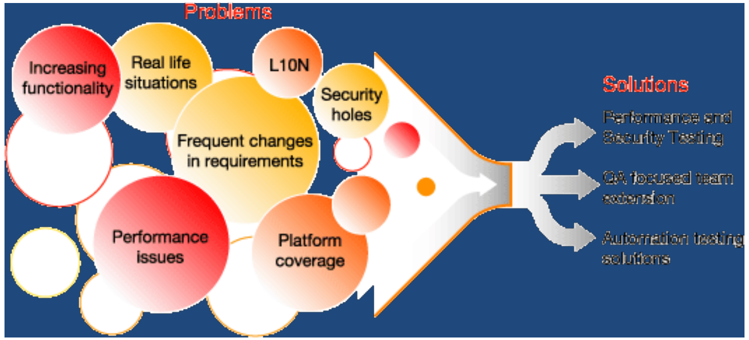
- You are expected to test your projects
  - To have a test plan
  - Test continuously (not just once)
- We will have testing labs next week
  - Each person will be responsible for testing some aspects of the project
  - With continuing responsibilities until the final hand-in
  - We will provide facilities for user testing among the class
  - We will be around to answer questions
- As we cover different testing methods
  - Think about how they affect your project

Getting a Project Ready to be Tested

- Want to have a back end 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 back end
  - Separate database
  - Add test users
  - Internal code to handle external actions
    - Based on which server is being run (e.g. switch databases)
  - Do it on a local machine / separate VM / separate port
  - Might have special URLs to reset the server to a known state
- Setup up test scripts
  - Reload the code, database, etc. to valid initial state

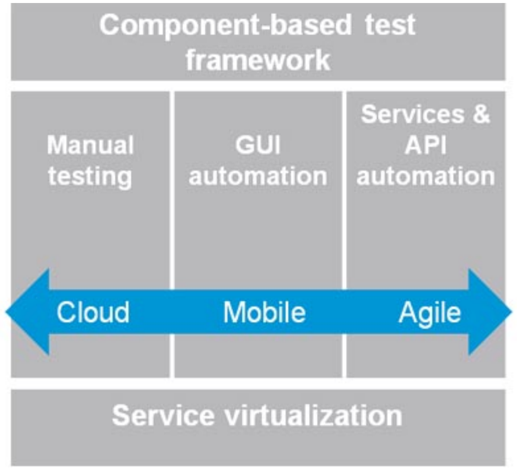
What Can Be Tested

- Usability (ease of use, speed of use, aesthetics, …)
- Accessibility
- Front end: HTML, CSS, Links
- Back end: unit test the node.js/php/python/java/…
- Application testing (front + back end)
- Browser Compatibility
- Performance
- Behavior under load/stress
- Security
- Internationalization, Printing, …


Problems
Increasing functionality
Real life situations
L10N
Security holes
Frequent changes in requirements
Performance issues
Platform coverage
Solutions
Performance and Security Testing
QA focused team extension
Automation testing solutions

Front End Testing

- Are the HTML and CSS correct?
  - What happens if they aren't?
- Does the site handle assistive devices?
- Can the site be crawled by search engines?
- Are the links correct (and active)
- Are the forms correct?
  - Are values validated correctly
  - Are default values correct
  - What happens to incorrect inputs
- Cookie testing
  - Does the application work without cookies
  - Are cookies encrypted correctly
  - Do sessions expire correctly



Front End Testing

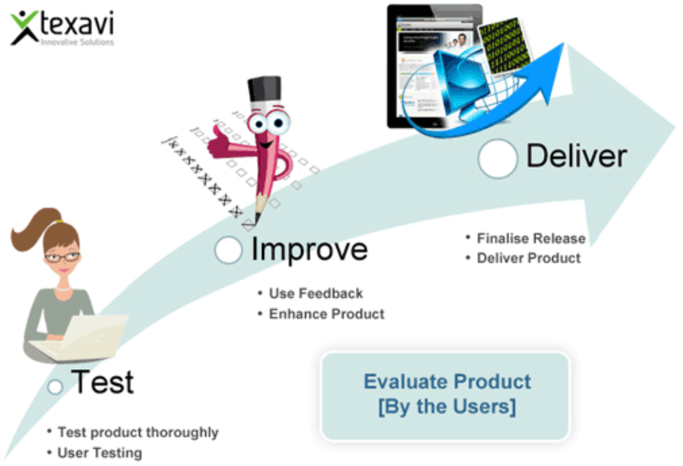
- HTML validation
  - W3C HTML validation service (http://validator.w3.org)
- Link checkers
  - W3C HTML validation (http://validator.w3.org/checklink)
- CSS validation
  - W3C CSS validation (http://jigsaw.w3.org/css-validator)
- Accessibility Testing
  - WAVE (https://wave.webaim.org)

Compatibility Testing

- Browser compatibility
  - IE, Firefox, Mozilla, Safari, Opera, Chrome, …
  - Different versions of each
  - Testing: browsershots.org
  - Testing: on-site testing
- OS Compatibility
  - What might be OS-dependent
- Mobile Compatibility
  - iPhone, Android, Blackberry, other phones
  - Different versions of each

User (Usability) Testing

- Test the effectiveness of the user interface
  - What is liked or disliked (subjective)
  - Speed and ease of use
  - What errors are made (and the error rate)
- How understandable is the interface
  - What instructions/help is required, what is obvious
- Is the content logical and easy to follow
  - Consistency of navigation and presentation
  - Spelling errors, colors and fonts, English
- Universal usability testing
  - Accessibility testing
  - Internationalization testing



Doing Usability Testing

- User studies
  - Watching users use the site (video taping for analysis)
  - Surveys or polls after use
  - Determining what information is needed
  - At least getting your friends or classmates to try it
- Log studies
  - What are the navigation paths? What are the common operations? How are key pages reached?
  - Detecting errors from the logs
  - Timings
  - Using Google Analytics and similar tools
- Tools and External Sources
  - http://www.youtube.com/watch?v=uLyWxXNDNbI
  - http://www.youtube.com/watch?v=xLIBe6VWmrY

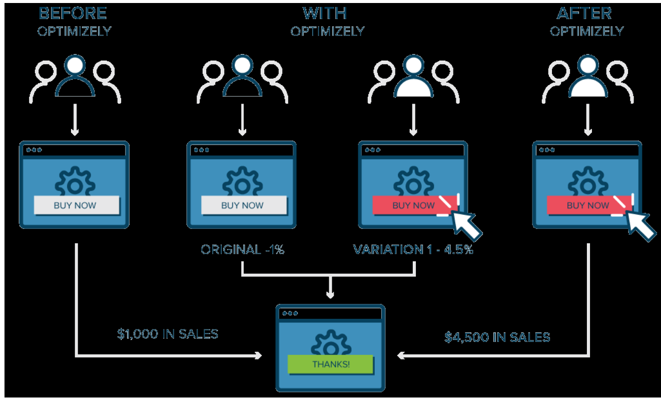
Usability Testing in CSCI1320

- We are providing user testing facilities
  - With you as guinea pigs
- Develop a well-thought out test first
  - What you want the user to do
  - What state the system should start in (and how to get there)
  - What questions you want to ask or have answered
- Create a starting page (optional)
  - Provide information needed to do the test, links to the test, etc.
  - Secret links to get the system in the right starting state
- Create a questionnaire form (web page)
  - Questions you want the user to answer, feedback desired
  - Google forms, qualtrics

Usability Testing in CSCI1320

- Define a test
  - Go to http://bdognom-v2.cs.brown.edu:5002
  - Do the user test lesson to add the data and produce a test
  - Keep it relatively simple (< 10 minutes of effort needed)
  - Only one active test per project at a time
  - Might want to try getting this ready by next week
- Take a test
  - Go to http://bdognom-v2.cs.brown.edu:5002
  - Do the take user test lesson to take a random test for another project
  - We'll do these next week in the testing labs

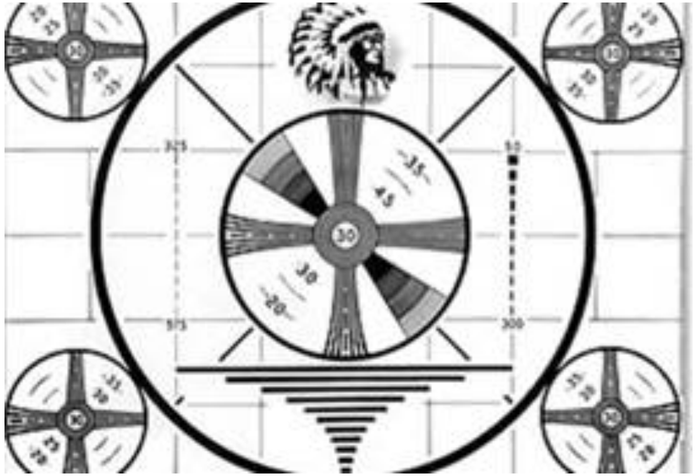
A/B User Testing



- Once you have a system working
  - Want to test possible modifications
- Randomly choose a subset of your users
  - Give them the new interface or new features
  - Give different subsets different new interfaces or features
  - Be consistent with each user
- Measure effectiveness, usability, etc.

Printing Testing

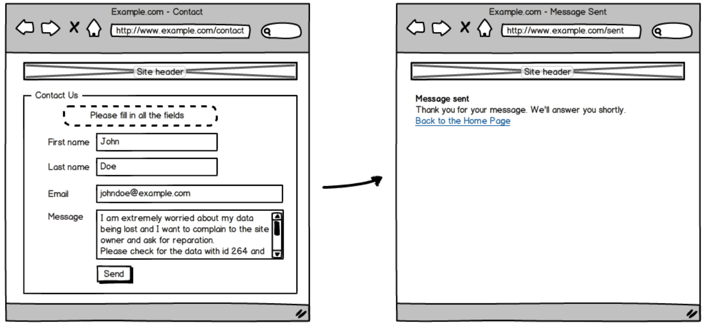
- Do the pages print correctly
  - Fonts, alignment
  - Size, layout
  - What prints, what doesn't (frames)
- Printing from different browsers
- Printing to different types of printers
- International printing



Front End Functional Testing

- Test the JavaScript in the front end
  - Unit testing (test functions individually)
  - Akin to using junit for Java
- Tools
  - Qunit, Jasmine
  - Introduction and examples:
    - http://qunitjs.com/intro/
  - Testing tools for jQuery
  - Several other JavaScript testing frameworks exist

Back End Functional Testing


Example.com - Contact

http://www.example.com/contact

Site header

Contact Us

Please fill in all the fields

First name: John

Last name: Doe

Email: johndoe@example.com

Message: I am extremely worried about my data being lost and I want to complain to the site owner and ask for reparation. Please check for the data with id 264 and

Send

Example.com - Message Sent

http://www.example.com/sent

Site header

Message sent

Thank you for your message. We'll answer you shortly.

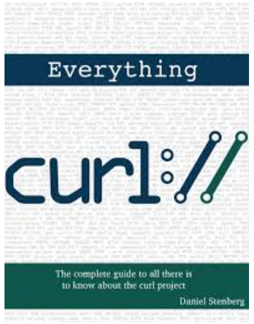
Back to the Home Page

- Does the back end code work?
  - Testing individual methods in the back end
- Are requests handled correctly?
  - Are the proper pages generated
  - Are the proper actions taken
- Depends on technology used in the back end
  - Simulate front end calls through function calls
  - Tools depend on language
- Tools:
  - Php: SimpleTest
  - Python/Django: PyUnit
  - Node (Jasmine, node-unit, Expresso, mocha + chai, nemo, Qunit, …)
    - See: http://jasmine.github.io/2.4/introduction.html
    - See: http://developer.android.com/training/testing/ui-testing/espresso-testing.html

Interface Testing

- What are the interfaces
  - Web page <-> web server
  - Web server <-> database
  - Web server <-> user server
- Check the interactions between these servers
  - Do they do the right thing
  - Are inputs validated properly
  - Are errors handled properly
  - Is validation and security correct
  - What happens if the user interrupts a transaction
  - What happens if the web connection is reset
  - What happens if the user clicks twice
- Testing the server code through its interfaces

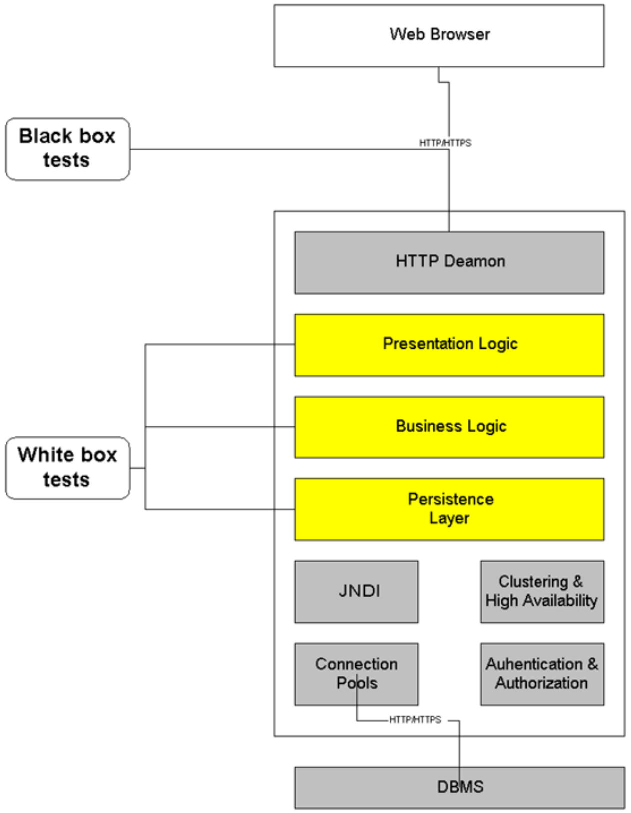
Interface Testing


Everything
curl://
The complete guide to all there is to know about the curl project
Daniel Stenberg

- 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 (for general scripts)
  - But you can put together scripts of curl calls to emulate tests

Interface 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
  - Tutorial: https://www.youtube.com/watch?v=rrVHoYBknGo
  - Overview: https://www.youtube.com/watch?v=01jFl8KrEMY
- Selenium
  - Next time



Next Time

- Testing web sites
- Homework:
  - Start getting your project web site or mobile app ready for testing
    - What should this entail
    - What do you need to do
    - Prelab for Wednesday

Question

What is not the purview of web site testing:

A. HTML, CSS, and Script validity
B. Determining if the SQL schema is consistent
C. Vulnerability to security attacks such as XSS
D. Universal accessibility
E. Determining how the web application handles a large user load

Usability Testing Tools

- UserTesting
  - http://info.usertesting.com/EduDemo.html
- Usage
  - Develop a well-thought out test first
    - What you want the user to do
    - What questions you want to ask
    - What questions you want answered
- Possible to actually use?
  - Sign up: https://www.usertesting.com/users/sign_up?client=true
  - Choose ORDER a TEST
  - Select no more than 3 participants
  - Need a code – ask the TAs if you are interested

Security

- What were you doing in the security lessons
  - Security is all about break web sites
- Suppose we wanted to fix and check again
  - Coming up with scripts that try to break it
  - Scripts that can be reused
  - Could you automate the hacking so it could be reused on other sites?

Security Challenge