

Objectives

- Application state
- Testing

May 10, 2016

Sprengle - CS335

1

Usability Review

- What are the key things to think about when considering usability?

May 10, 2016

Sprengle - CS335

2

APPLICATION STATE

May 10, 2016

Sprengle - CS335

3

Web Application State So Far

- Where are the various places that we can store information/state?
- What information should be kept there?
- How do we access that information?

May 10, 2016

Sprengle - CS335

4

Web Application State So Far

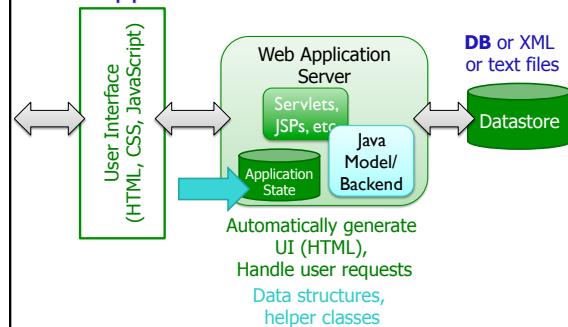
- Parameters
 - Client to server
 - Server: `request.getParameter()`
- Request attributes
 - Server; servlet/jsp communication
 - Server: `request.getAttribute()`
- Session attributes
 - Server; Information about one user
 - Server: `session.getAttribute()`

May 10, 2016

Sprengle - CS335

5

Web Application Architecture Overview



May 10, 2016

Sprengle - CS335

6

ServletContextListener

- Sets up `ServletContext`
 - What all servlets in web app can see
- `ServletContextListener`'s `contextInitialized` called when application is first initialized
 - Before any servlets are initialized
- `ServletContextListener`'s `contextDestroyed` called when application is shutdown



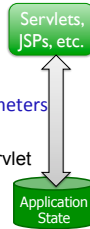
May 10, 2016

Sprengle - CS335

7

ServletContextListener Example: DBCPoolingListener

- `contextInitialized` creates an **application** attribute: `DBCPool`
 - DB Connection Pool
 - Object to get information from the DB
 - Creates DB connection pool using context parameters
 - Context parameters are defined in web.xml
 - Access by calling `getInitParameter` on servlet context object
- Servlets access using `ServletContext`
- JSPs access using `application` variable



May 10, 2016

Sprengle - CS335

8

WEB APPLICATION TESTING

May 10, 2016

Sprengle - CS335

9

Why Test Web Applications?

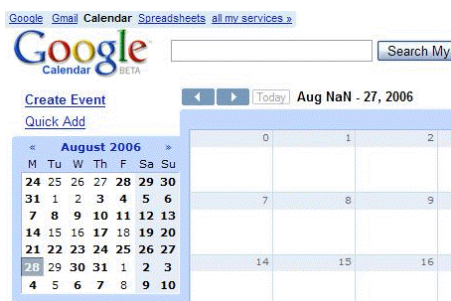


May 10, 2016

Sprengle - CS335

10

Why Test Web Applications?



May 10, 2016

Sprengle - CS335

11

Helping the Japanese TA



May 10, 2016

Sprengle - CS335

12

More Errors

Thank you, , for your generous donation to Toys for Tots.

Renewal Cancellation Confirmation

Your renewal to
Fatal error: Call to a member function name() on a non-object in C:\Repository\weblib\uedolphin-magazines.com\templates\smarts\subcontent\account_history_cancel_confirm.tpl.php on line 21

Error in connection file "/private/var/folders/qr/r15h4h_x5vb69r24khg97pjw0000pp/T/Citrix_00.ica": Cannot find section "ApplicationServers".
 Error number: 11

quiznos near 3401 Atlantic Avenue, Virginia Beach, VA 23451-2854

Quiznos Sandwich Restaurants
 1800 Skibo Rd Ste 252, Fayetteville, NC 1.1 mi S
 (910) 826-0840 - quiznos.com

May 10, 2016 Spenkle - CS335 13

Other Errors

We're sorry

BILLING INFORMATION

* Required
 You will have the ability to edit this information later in the process.

How would you like to pay?

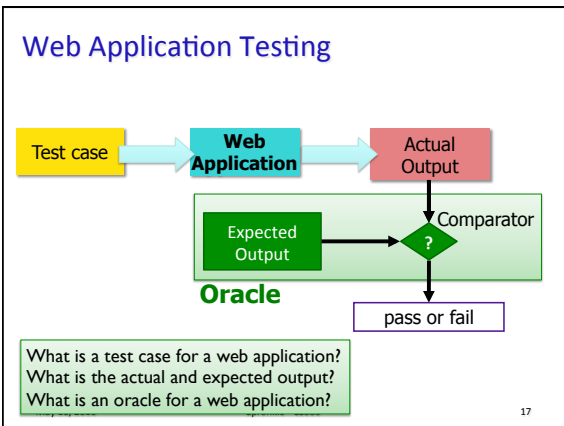
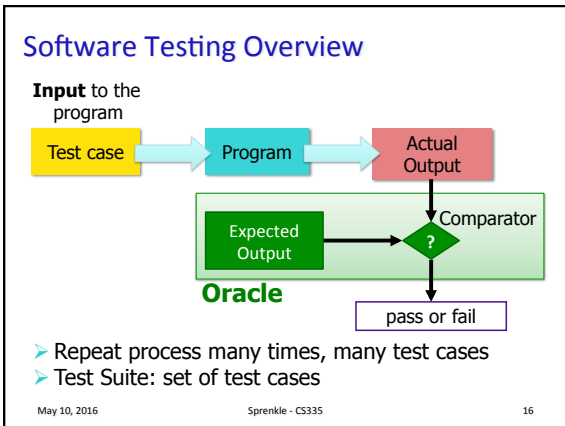
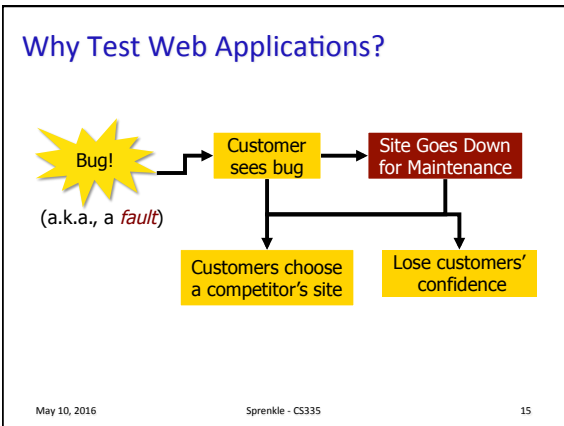
* Required

GIFT CARD CLOSE X

UNABLE TO ACCESS BALAN The available balance on gift card #60...732 is \$50.0.

Credit Card

May 10, 2016 Spenkle - CS335 14



- ### Web Application Testing
- What are typical approaches to testing?
 - How do they apply to web application testing?
 - What are the challenges in testing web applications?
 - For you?
 - For professional developers?
- May 10, 2016 Spenkle - CS335 18

Web Application Testing Challenges

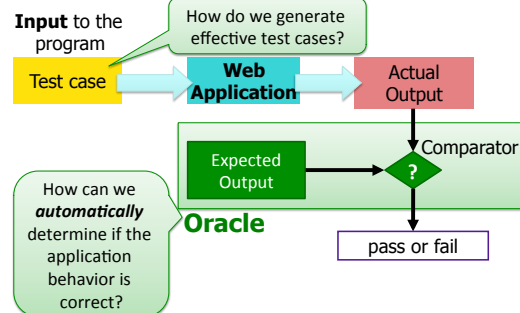
- Automation
 - Maintain availability
 - Frequent application updates
 - Regression testing
 - Cross-browser compatibility
 - Security demands
 - Usability testing
 - Appearance
 - Concurrent users
 - Generating "good" test cases
 - Determining correctness of application
- Some of these are not unique to web applications

May 10, 2016

Sprengle - CS335

19

Web Application Testing Challenges

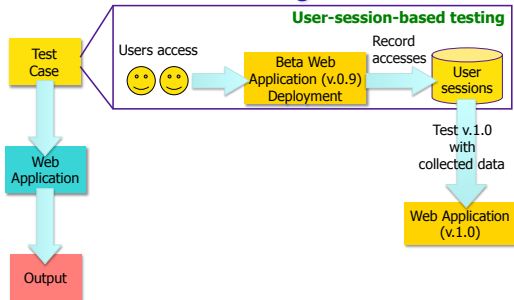


May 10, 2016

Sprengle - CS335

20

One Approach: User-Session-Based Testing

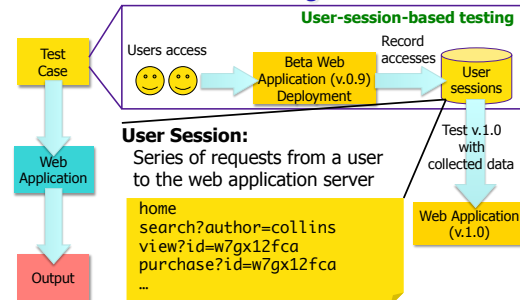


May 10, 2016

Sprengle - CS335

21

One Approach: User-Session-Based Testing

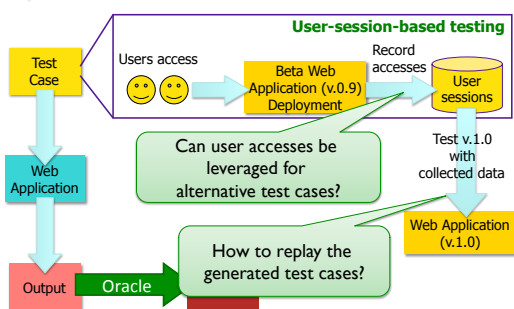


May 10, 2016

Sprengle - CS335

22

My Dissertation Focus

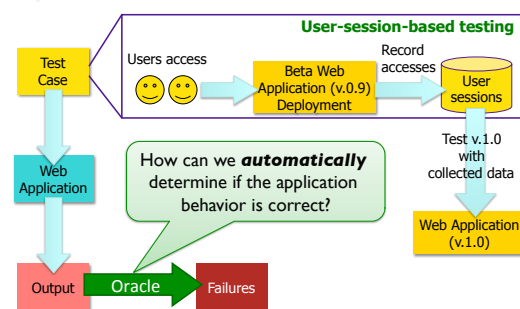


May 10, 2016

Sprengle - CS335

23

My Dissertation Focus



May 10, 2016

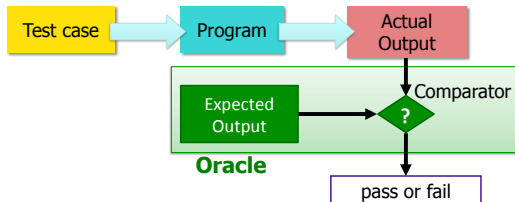
Sprengle - CS335

24

Background: Oracles

Oracle: Quality Inspector

- Determine if test cases pass, i.e., program behaves as expected for that input
- If does not pass, indicates a fault/bug in application

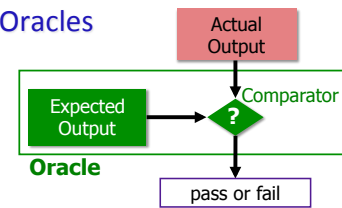


May 10, 2016

Sprengle - CS335

25

Background: Oracles



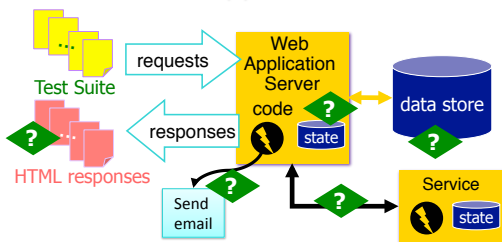
- Automatically generating **expected** output
 - specification
 - trusted implementation
 - simulations
 - domain-specific model
- Comparator: validating **actual** output

May 10, 2016

Sprengle - CS335

26

Oracles for Web Applications



- Challenge: **Low observability** of outputs
- Solution: Use "pseudo-oracle" [Weyuker 82] on HTML
 - HTML: Common output for web applications, users see output, side effects of failures may show up in HTML

May 10, 2016

Sprengle - CS335

27

Alternative Approach: Sikuli

- Takes screenshots of your GUI (e.g., browser) at specified times
- Compares the image of the actual and expected output

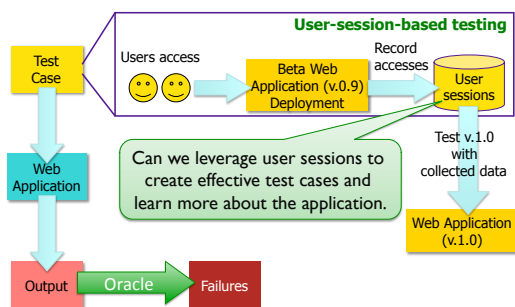
<http://www.sikuli.org/>

May 10, 2016

Sprengle - CS335

28

Focus of W&L Web Application Testing Lab

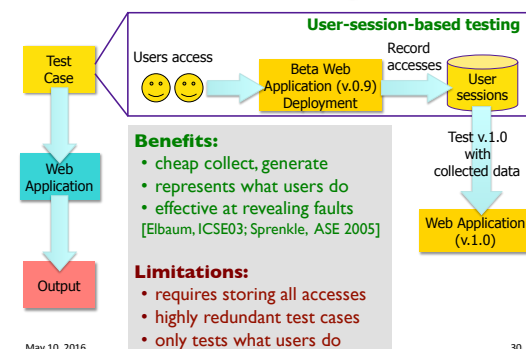


May 10, 2016

Sprengle - CS335

29

User-Session-Based Testing



May 10, 2016

Sprengle - CS335

30

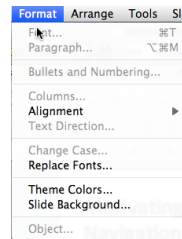
Issues in Generating Test Cases for Web Applications

- Need to represent
 - Application *navigation*, e.g., sequence of requested pages
 - Data entered, e.g., through forms
- Goals
 - Expose *faults* (i.e., bugs)
 - **Low generation and execution cost**

ICST2011, ICST2012

Navigating Applications

Graphical User Interface (GUI)



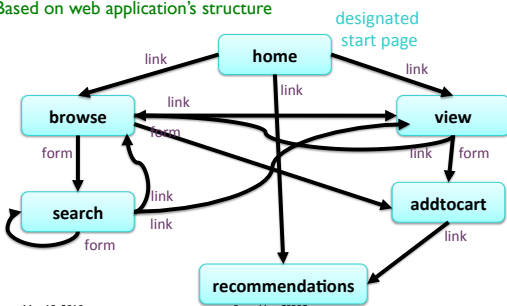
Web Application

- Developer may implement restrictions
- Browser navigation
 - Back, forward buttons
 - Location bar
 - Multiple windows
- In a 2004 study, 47% of user sessions included an "infeasible" navigation*

* P. Tonella and F. Ricca. "Statistical testing of web applications." *Journal of Software Maintenance and Evolution*, 16(1-2):103-127, 2004.

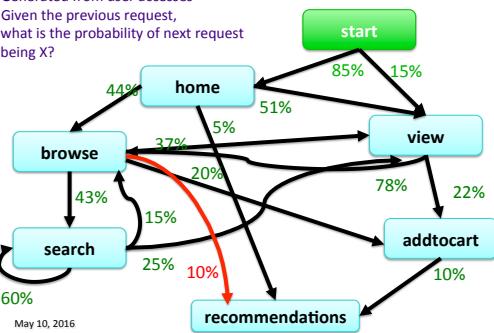
Example Simplified Intra-session Navigation Model

Based on web application's structure

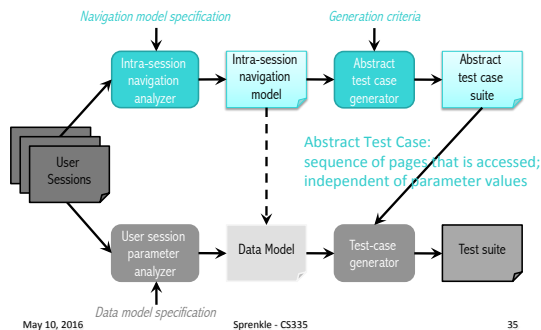


Sant et al. Model, n=2

- Generated from user accesses
- Given the previous request, what is the probability of next request being X?



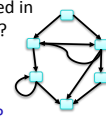
Test Case Generation Process



Research Questions:

Factors in Constructing Navigation Model

- How should **requests** be represented?
- How much **history** (i.e., what value of n) should be used in the n-gram Markov model to predict the next request?
- How should **transitions** between user requests be approximated?
 - How should edges in navigation model be labeled?
- How **many user sessions** should be used to build the model?
- How does **model size** vary with different configurations for building the model?



Research Questions: Generated Abstract Test Cases

- How representative are the abstract test cases of the original user sessions?
- Do the abstract test cases enable testing of usage not found in the user sessions?

May 10, 2016

Sprengle - CS335

37

Research Questions: User-specific models

- Do different users access a web application differently?
- Can we automatically identify different types of users and create different models for those users?

May 10, 2016

Sprengle - CS335

38

More Open Questions/Topics

- TC Generation: Data model!
 - Need to create *executable* test cases
- Generating a test suite
 - Order of executing test suite
- What else can we learn from the navigation model? Data model?
- Do these approaches work well for web services?
 - How to adapt the approaches?
- Oracles!

May 10, 2016

Sprengle - CS335

39

UNIT TESTING - WEB

May 10, 2016

Sprengle - CS335

40

HtmlUnit



- Extension to JUnit
 - Test individual units
 - Write code as in JUnit
- Write tests, assertions manually
 - Run tests multiple times, e.g., after each nightly build
- Runs in its own emulated browser
 - No visual output, not as good with JavaScript
- JavaScript support
- Built on by other frameworks
<http://htmlunit.sourceforge.net/>

May 10, 2016

Sprengle - CS335

41

HtmlUnit Example Code

```
@Test
public void homePage() throws Exception {
    final WebClient webClient = new WebClient();
    try (final WebClient webClient = new WebClient()) {
        final HtmlPage page = webClient.getPage(
            "http://htmlunit.sourceforge.net");
        Assert.assertEquals("HtmlUnit - Welcome to HtmlUnit",
            page.getTitleText());
        final String pageAsXml = page.asXml();
        Assert.assertTrue(pageAsXml.contains(
            "body class=\"composite\""));
        final String pageAsText = page.asText();
        Assert.assertTrue(pageAsText.contains(
            "Support for the HTTP and HTTPS protocols"));
    }
}
```

May 10, 2016

Sprengle - CS335

42

HtmlUnit Example Code

```
@Test
public void submittingForm() throws Exception {
    try (final WebClient webClient = new WebClient()) {
        // Get the first page
        final HtmlPage page1 = webClient.getPage("http://some_url");
        // Get the form that we are dealing with and within that form,
        // find the submit button and the field that we want to change.
        final HtmlForm form = page1.getFormByName("myform");
        final HtmlSubmitInput button =
            form.getInputByName("submitButton");
        final HtmlTextInput textField = form.getInputByName("userid");
        // Change the value of the text field
        textField.setValueAttribute("root");
        // Now submit the form by clicking the button and get back the
        // second page.
        final HtmlPage page2 = button.click();
    }
}
```

May 10, 2016

Sprengle - CS335

43

Canoo's WebTest

- Uses HtmlUnit as underlying "browser"
- Write tests in XML/Ant
- Ant: Java-based build tool
 - Similar to Make (without the wrinkles)
 - Used explicitly inside of NetBeans IDE
- Generates better reports than the JUnit style
 - Easier to understand why web app fails

May 10, 2016

Sprengle - CS335

44

Example WebTest Script

```
<project default="test">
  <target name="test">
    <webtest
      name="check that WebTest is Google's top 'WebTest'
      result">
      <invoke url="http://www.google.com/ncr"
        description="Go to Google (in English)"/>
      <verifyTitle text="Google" />
      <setInputField name="q" value="WebTest" />
      <clickButton label="I'm Feeling Lucky" />
      <verifyTitle text="Canoo WebTest" />
    </webtest>
  </target>
</project>
```

May 10, 2016

Sprengle - CS335

45

Many Other Similar Testing Tools

- JWebUnit
- WebDriver
- JSFUnit
- Wetator
- Celerity
- Spring MVC Test HtmlUnit
- ...

May 10, 2016

Sprengle - CS335

46

Selenium

Demonstration
Map to earlier testing framework

- Capture-replay tool
- Runs test cases in browser
 - Easy to see results
- Save test cases in test suites
 - Rerun test cases multiple times
- Can assert various attributes to be true (or false) to determine if test case passes
 - Manually written oracles
 - Assert vs Verify
- Locations:
 - Specified using DOM, XPath, or id

May 10, 2016

Sprengle - CS335

47

Exam Questions?

May 10, 2016

Sprengle - CS335

48

TODO

- Lab 8 - Testing Practice
 - Create test cases with Selenium – due tonight at midnight
- Analysis of Bugs
 - Keep track of your bugs, their causes, locations, how you found them, ...
 - Stored in Jira
 - Due Friday, May 20
- Wed midnight – high priority functionality
- Exam – Thursday
 - Prep document available

May 10, 2016

Sprenkle - CS335

49