Homework Peer Review

Critical to teaching:

- feedback management
  - particularly peer review management
  - anonymous and volunteer reviewing
    - in a public school environment there are a number of adults who could help with the review process as long as the document creator/ writer was made anonymous.
  - peer review
    - critical to maintain teacher sanity is a way to 'farm off' paper reviews to peers... so making viewing of assignments anonymous is very important... there's no human way to read 100+ papers in a reasonable amount of time

The desired feedback mechanisms would be form based and generated in the Tikiwiki environment for

- history tracking
- reuse of templates
  - modifying the tiki_parse might be required... being able to upload pages for parsing into wiki surveys, faqs, trackers, quizzes etc. is a huge step towards making tiki a speedier content management dev tool
- transferability

<textarea cols="80" rows="5" name="text area" color="red">Of critical importance is that a student cannot begin reviewing other student's work without first having submitted their own... this lessens the possibility that a student cheats.</textarea>

The text areas need to a have a js to count up the number of words... students will just type a couple of letters and then submit so that they can go look at other student's work for ideas and etc.

The wiki tags would be converted to html forms fields... mose has indicated he has begun work on this idea. (I need to identify which fields are supported in wiki pages...)
---
here's the short answer fill in the blank
<form name="test_form"><input name="testfield" value="Wiki form Builder"></form>

---
here's my short answer/ essay field
Here's a working text area... the tags required for creating this text area field could be created using QuickTags (unfortunately, right now, there's a bug in the QuickTags that doesn't allow more than 21 quick tags... other wise I'd create demos of all of these tags...
<textarea cols="80" rows="5" name="text area">Instructions for the assignment can go in here or templates for the students to use while writing?</textarea>

---
need those rolldowns:
I lifted this straight from the W3C site on HTML forms... mozilla's composer under win2k is buggy, preview doesn't work.
<select name="Verb">
<option selected="selected" label="none" value="none">Click the rolldown</option>

<optgroup label="Verbs">
<option label="3.7.1" value="pm3_3.7.1">Past Tense</option>
<option label="3.7" value="pm3_3.7">Present</option>
<option label="3.5" value="pm3_3.5">Future</option>
</optgroup><optgroup label="Adjectives">
<option label="3.7" value="pm2_3.7">green</option>
<option label="3.5" value="pm2_3.5">yellow</option>
</optgroup><optgroup label="Averbs">
<option label="3.7R" value="IRX_3.7R">quickly</option>
<option label="3.5R" value="IRX_3.5R">slowly</option>
</optgroup>
</select>

</p>

Imagine having to retype options like this one. This is why the QuickTags can be very useful and why the use of wiki form templates are critical for speeding development and delivery of content and assignments. Ideally there would be a way of knowing what templates are being used at any one time but that's a future problem.
<select name=grade >
<option selected="selected" label="none" value="none">Assign point value for this test...</option>
<option value="100" selected>100</option>
<option value="99">99</option>
<option value="98">98</option>
<option value="97">97</option>
<option value="96">96</option>
<option value="95">95</option>
<option value="94">94</option>

<option value="93">93</option>
<option value="92">92</option>
<option value="91">91</option>
<option value="90">90</option>
<option value="89">89</option>
<option value="88">88</option>

<option value="87">87</option>
<option value="86">86</option>
<option value="85">85</option>
<option value="84">84</option>
<option value="83">83</option>
<option value="82">82</option>

<option value="81">81</option>
<option value="80">80</option>
<option value="79">79</option>
<option value="78">78</option>
<option value="77">77</option>
<option value="76">76</option>

<option value="75">75</option>
<option value="74">74</option>
<option value="73">73</option>
<option value="72">72</option>

<option value="71">71</option>
<option value="70">70</option>

<option value="69">69</option>
<option value="68">68</option>
<option value="67">67</option>
<option value="66">66</option>
<option value="65">65</option>
<option value="64">64</option>

<option value="63">63</option>
<option value="62">62</option>
<option value="61">61</option>
<option value="60">60</option>
<option value="59">59</option>
<option value="58">58</option>

<option value="57">57</option>
<option value="56">56</option>
<option value="55">55</option>
<option value="54">54</option>
<option value="53">53</option>
<option value="52">52</option>

<option value="51">51</option>
<option value="50">50</option>
<option value="49">49</option>
<option value="48">48</option>
<option value="47">47</option>
<option value="46">46</option>

<option value="45">45</option>
<option value="44">44</option>
<option value="43">43</option>
<option value="42">42</option>
<option value="41">41</option>
<option value="40">40</option>

<option value="39">39</option>
<option value="38">38</option>
<option value="37">37</option>
<option value="36">36</option>
<option value="35">35</option>
<option value="34">34</option>

<option value="33">33</option>
<option value="32">32</option>
<option value="31">31</option>
<option value="30">30</option>
<option value="29">29</option>
<option value="28">28</option>

<option value="27">27</option>
<option value="26">26</option>
<option value="25">25</option>

<option value="24">24</option>
<option value="23">23</option>
<option value="22">22</option>

<option value="21">21</option>
<option value="20">20</option>
<option value="19">19</option>
<option value="18">18</option>
<option value="17">17</option>
<option value="16">16</option>

<option value="15">15</option>
<option value="14">14</option>
<option value="13">13</option>
<option value="12">12</option>
<option value="11">11</option>
<option value="10">10</option>

<option value="9">9</option>
<option value="8">8</option>
<option value="7">7</option>
<option value="6">6</option>
<option value="5">5</option>
<option value="4">4</option>

<option value="3">3</option>
<option value="2">2</option>
<option value="1">1</option>
<option value="0">0</option>
</select>
---
these fields could be supplied from a simple text or wiki file like this:

```
i = 'cat yourrubricfile.txt';
for n in i;
do grep $n form.txt;
done
```

Or maybe not... just a thought

Check boxes are good in forms

<input type="checkbox" name="this checkbox" value="sessions">click here	<input type="checkbox" name="this checkbox" value="sessions">	or here<input type="checkbox" name="this checkbox" value="sessions">	or there<input type="checkbox" name="this checkbox" value="sessions">

And how could we tune our pages to the right tone without radio buttons?
<input type="radio" name="grade0" value="1">
---
or perhaps you want to type in your own value?
here's the short answer fill in the blank
<form name="test_form"><input name="testfield" value="Type the value of this assessment"></form>

the problem of submitting from the wiki interface is difficult but the idea is not that the wiki page is the holder of the form... the form interface reads from the wiki page to generate the form interface...

Q: More tags? Isn't that tough to manage and remember?
A: No, the tag data can be saved in mose QuickTags for easy customizataion.

Q: What kind of tags would you need?
A: Basic ones like:
{Title: (form type=text area, text box, etc | #description/ instructions}
each field should allow the teacher to give specific and easily modified instructions over the field... teachers have to adjust documents in real time quickly and easily... that's why modifying a form in the wiki environment is almost ideal.
Description:
Formatting:
Maximum grade:
Grading Strategy:
Number of Comments, Assessment Elements, Grade Bands, Criterion Statments or Categories in a Rubric:
Allow Resubmissions:
Number of Assessments of Examples from Teacher:
Number of Assessments of Student Submissions:
Self Assessment:
Assessments must be agreed:
Hide Grades before Agreement:
Maximum Size:
Deadline:

Don't want

file upload without strict control of mime types is a BAD thing

- uploading virus infected files like .doc or .ppt onto a teacher's school machine is a bad thing... and not all students have the software so keep it simple: force .txt and .html as file format.

Specifications Please

George Geller says: Homework Peer Review seems to be a real can of worms. For example: Who willl review the reviews? How many other students' submissions is one student allowed to review? How many reviews must a submission get before it goes to the teacher or for re-editing? Before I can even start to think about coding Homework Peer Review I need to know what items will be on the screens and what they will do. If you could generate these specs, it would speed thing up down the road.