

Navigation around the system is inconsistent and slow

Because of the original URL file was crowded, when it should have been spread out into different URL files in different folders it caused the navigation to be inconsistent and slow as django would find the first occurrence of a file, not the correct one.

By spreading the URL's out into multiple files it solved the issues with navigation.

Once this had been solved, the navigation was correct for all pages of the system.

Testing this will be important to ensure all of the navigation is consistent and therefore easy for users to be able to find the correct resources.

Fixed Edit answer.html forms

On the html page called 'Edit_answered.html' whenever the csrf was passed, it would trigger an error, to be able to fix this we changed render_to_response which fixed this error.

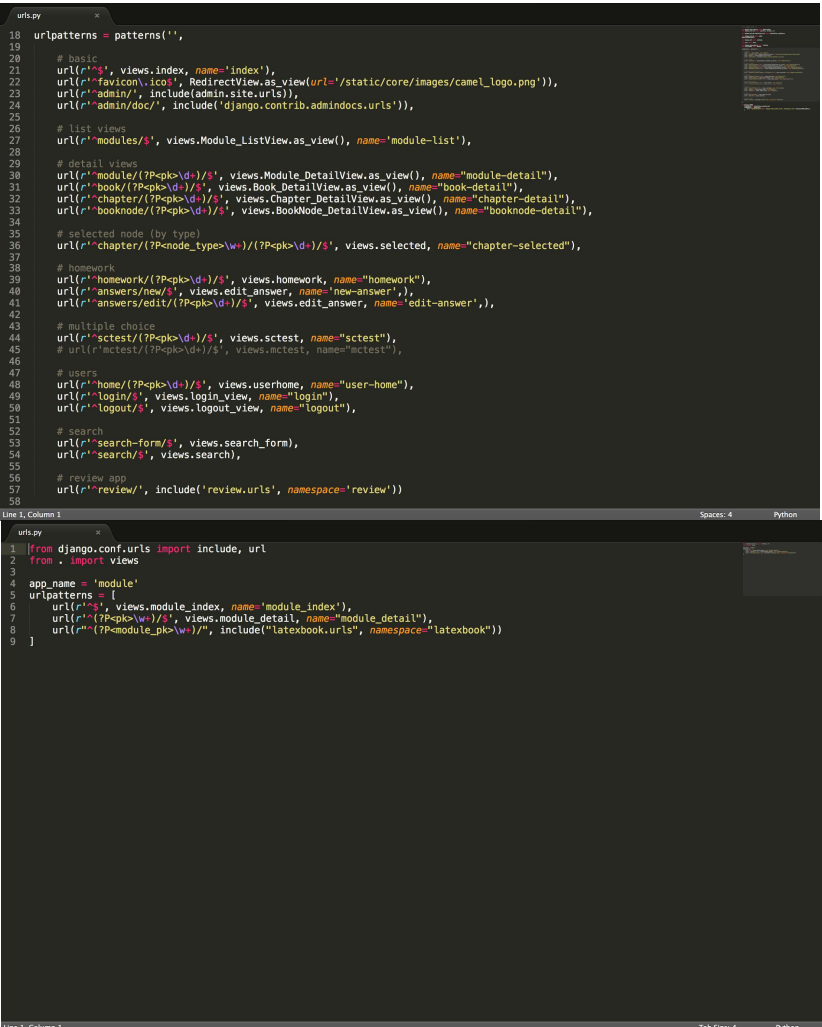
Fixed broken navigation link issue

Due to multiple templates with the same name but for different applications in the URL's folder, it was returning the first URL it came across, not the correct one. By splitting these up into separate files (namespacing) this has solved the issues of broken links.

Student answers error handling

When a student entered incorrect latex into an input box and tried to save, it would throw errors ungracefully. We have added jquery to be able to gracefully handle the situation to stop students submitting empty or invalid inputs when attempting to save.

This would be important to test as when this system is being used, if errors are not thrown and faulty latex is submitted, this can cause users grades to not reflect their knowledge fully as it may be un-markable.



```
18 urlpatterns = patterns('',
19
20     # basic
21     url(r'^$', views.index, name='index'),
22     url(r'^favicon.ico$', RedirectView.as_view(url='/static/core/images/camel_logo.png')),
23     url(r'^admin/', include(admin.site.urls)),
24     url(r'^admin/doc/', include('django.contrib.admindocs.urls')),
25
26     # list views
27     url(r'^modules/$', views.Module_ListView.as_view(), name='module-list'),
28
29     # detail views
30     url(r'^module/(?P<pk>\d+)/$', views.Module_DetailView.as_view(), name='module-detail'),
31     url(r'^book/(?P<pk>\d+)/$', views.Book_DetailView.as_view(), name='book-detail'),
32     url(r'^chapter/(?P<pk>\d+)/$', views.Chapter_DetailView.as_view(), name='chapter-detail'),
33     url(r'^booknode/(?P<pk>\d+)/$', views.BookNode_DetailView.as_view(), name='booknode-detail'),
34
35     # selected node (by type)
36     url(r'^chapter/(?P<node_type>\w+)/(?P<pk>\d+)/$', views.selected, name='chapter-selected'),
37
38     # homework
39     url(r'^homework/(?P<pk>\d+)/$', views.homework, name='homework'),
40     url(r'^answers/new/$', views.edit_answer, name='new-answer'),
41     url(r'^answers/edit/(?P<pk>\d+)/$', views.edit_answer, name='edit-answer'),
42
43     # multiple choice
44     url(r'^sctest/(?P<pk>\d+)/$', views.sctest, name='sctest'),
45     # url(r'^mctest/(?P<pk>\d+)/$', views.mctest, name='mctest'),
46
47     # users
48     url(r'^home/(?P<pk>\d+)/$', views.userhome, name='user-home'),
49     url(r'^login/$', views.login_view, name='login'),
50     url(r'^logout/$', views.logout_view, name='logout'),
51
52     # search
53     url(r'^search-form/$', views.search_form),
54     url(r'^search/$', views.search),
55
56     # review app
57     url(r'^review/', include('review.urls', namespace='review'))
58 )
```

```
1 from django.conf.urls import include, url
2 from . import views
3
4 app_name = 'module'
5 urlpatterns = [
6     url(r'^$', views.module_index, name='module_index'),
7     url(r'^(?P<pk>\w+)/$', views.module_detail, name='module_detail'),
8     url(r'^/(?P<module_pk>\w+)/$', include('latexbook.urls', namespace='latexbook'))
9 ]
```

Fixed Mathjax

when a student saved data into the system, the mathjax didn't update itself meaning that the inputted latex would not be displayed on the screen on submission. By adding new code in the mathjax now updates itself automatically, allowing the students to see the latex which they had just input.

This will be important to test as if the latex is not displayed properly, it would look unprofessional for the university. It would also have the effect that users may not know if the latex they have input is correct, and may be submitting faulty answers without knowing.

latex parser bug

The latex parser prevented the MPTT from accessing the children in the modules tree, therefore not displaying correctly on the system. This has now been solved by fixing faulty code.

We would have to test this so that the modules, tests, books, etc. will have be displayed properly on the system, so that the students can access the correct handouts and complete the correct tests.

Parser being unaware of a missing bracket

We've overhauled the parser so that it checks the latex for uneven parenthesis, and throws errors gracefully instead of crashing.

Original file structure Vs new file structure

The original file structure of the system was messy and relatively unorganised. Files were scattered all over the place. It became very hard to manage and thus pinpointing what each file was responsible for during testing would quickly become a harrowing task...

As a result of this we decided it would be easier and more efficient if we cleaned it up and modularised the code. This enabled us to grasp a better understanding of the code and makes testing considerably easier to accomplish. Furthermore this allows future maintenance to be carried out more effectively as it is clearer as to what everything does.

