MVP2-Modularity Project:

The goal of this small refractor is to make use to Django decoupling capabilities to make the templates more pluggable. This goes over a daily report of changes made, how to use them, and examples of their use. Most modules can be found in the “templates” folder of “VSB”.

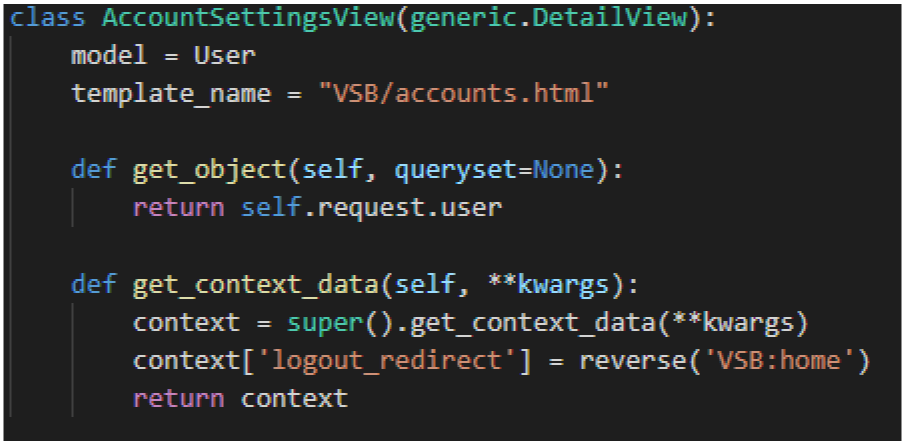
10/13/2020

Pluggable Login Buttons

Two buttons were added, “VSB/modules/login-button-inplace.html” and “VSB/modules/login-button.” These do log-in, log-out easily.

Inplace login will not change the url of the user before login nor after logout.

The standard button requires the ‘logout_redirect’ context variable to be defined. You can see this in AccountSettingsView.



```
class AccountSettingsView(generic.DetailView):
    model = User
    template_name = "VSB/accounts.html"

    def get_object(self, queryset=None):
        return self.request.user

    def get_context_data(self, **kwargs):
        context = super().get_context_data(**kwargs)
        context['logout_redirect'] = reverse('VSB:home')
        return context
```

Pluggable Profile Top Bar

The top bar seen below is now modular and can be easily added.

To do so, simply include

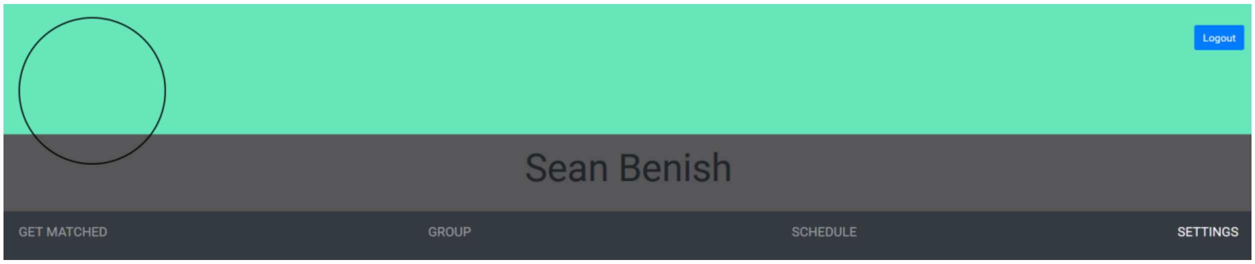
```
{% include ‘VSB/modules/profile-topbar.html’ %}
```

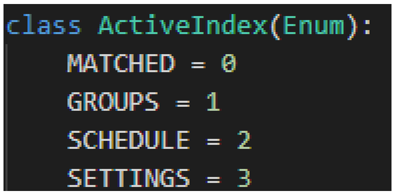
where the topbar should go and place

```
<link rel="stylesheet" href="{% static 'VSB/profile-topbar.css' %}>
```

in the header. You can see “VSB/settings.html” for an example.

Note, the active (highlighted) item in the navbar uses the context variable ‘active_index’. To see values for this variable, see the ‘ActiveIndex’ enum in ‘VSB/views.py’.





```
class ActiveIndex(Enum):
    MATCHED = 0
    GROUPS = 1
    SCHEDULE = 2
    SETTINGS = 3
```