Authentication views

https://github.com/chezmarcbrown/classdemo-2022-11-17-authorization.git

> login/logout (django.contrib.auth)

auctions/urls.py

```
from django.urls import path
from django.contrib.auth import views as auth_views

from . import views

urlpatterns = [
    path("", views.index, name="index"),
    #path("login", views.login_view, name="login"),
    #path("logout", views.logout_view, name="logout"),
    path("register", views.register, name="register"),

    path('login', auth_views.LoginView.as_view(), name='login'),
    path('logout', auth_views.LogoutView.as_view(), name='logout'),
]
```

Looks for registration/login.html

---

auctions/templates/registration/login.html

```
{% extends "auctions/layout.html" %}
{% block main %}
```

```
    <h2>Login</h2>

    <form method="post">
        {% csrf_token %}
        {{ form.as_p }}
        <input type="submit" value="Login">
    </form>

    <p>Don't have an account? <a href="{% url 'register' %}">Register here.</a></p>

{% endblock %}
```

What happens after successful login? By default, looks for route “profile”. Can be changed in settings.py:

```
LOGIN_REDIRECT_URL = "/"
```

Logout just works, but needs to say where to redirect… So, change settings.py:

```
LOGOUT_REDIRECT_URL = "/"
```

> Change Password (django.contrib.auth)

Add routes to urls.py

```
urlpatterns = [
    . . .
    path('password_change', auth_views.PasswordChangeView.as_view(), name='password_change'),
    path('password_change/done', auth_views.PasswordChangeDoneView.as_view(), name='password_change_done'),
]
```

Add button to layout.html

```
{% if user.is_authenticated %}
    Welcome <strong>{{ user.username }}!</strong>
    <a href="{% url 'password_change' %}">Change Password</a>
    <a href="{% url 'logout' %}">Log Out</a>
```

Default templates exist, but can provide app-specific.

registration/password_change_form.html:

```
{% extends "auctions/layout.html" %}
{% block main %}

    <h2>Change Password</h2>

    <form method="post">
        {% csrf_token %}
        {{ form.as_p }}
        <input type="submit" value="Change">
    </form>

{% endblock %}
```

registration/password_change_done.html:

```
{% extends "auctions/layout.html" %}
{% block main %}

    <h2>Password has been changed!</h2>

{% endblock %}
```

Recall that password validators controlled in settings.py:

```
AUTH_PASSWORD_VALIDATORS = [...]
```

> Reset Password (django.contrib.auth)

```
# - PasswordResetView sends the mail with a link for user to click
# - PasswordResetDoneView shows a success message for the above
# - PasswordResetConfirmView checks the link the user clicked and prompts for a new password
# - PasswordResetCompleteView shows a success message for the above
```

Add routes to urls.py

```
path('password_reset/', auth_views.PasswordResetView.as_view(), name='password_reset'),
path('password_reset/done/', auth_views.PasswordResetDoneView.as_view(), name='password_reset_done'),
path('reset/<uidb64>/<token>/', auth_views.PasswordResetConfirmView.as_view(), name='password_reset_confirm'),
path('reset/done/', auth_views.PasswordResetCompleteView.as_view(), name='password_reset_complete'),
```

Add “Forgot password?” to login.html

```
<p>Forgot your password? <a href="{% url 'password_reset' %}">Send me email.</a></p>
<p>Don't have an account? <a href="{% url 'register' %}">Register here.</a></p>
```

Configure an email backend to just store email in a file in settings.py:

```
EMAIL_BACKEND = "django.core.mail.backends.filebased.EmailBackend"
EMAIL_FILE_PATH = os.path.join(BASE_DIR, 'sent_emails')
```

Forms that can be customized:

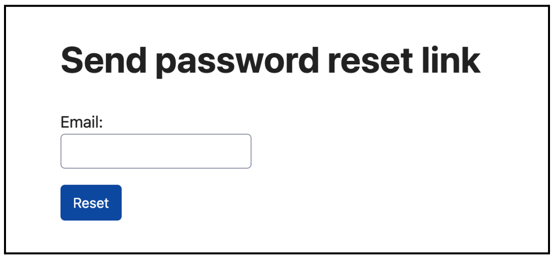
Password_reset_email.html - body of email that will be sent
Password_reset_subject.txt - the Subject field of email that will be sent‹


sent_emails > 20221117-163202-4472757056.log

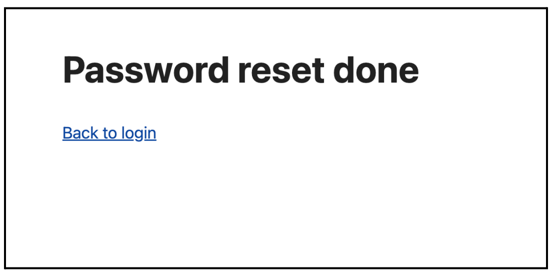
```
Content-Type: text/plain; charset="utf-8"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Subject: Password Reset Request for UAA490
From: webmaster@localhost
To: joe@foo.com
Date: Thu, 17 Nov 2022 16:32:02 -0000
Message-ID:
 <166870272257.31364.12950197983851771012@1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.

Someone asked for password reset for email joe@foo.com.
Follow the link below:
http://127.0.0.1:8000/reset/Mg/bf1kle-5b9cbb698c091ed8f8788037ceee65e7/
```

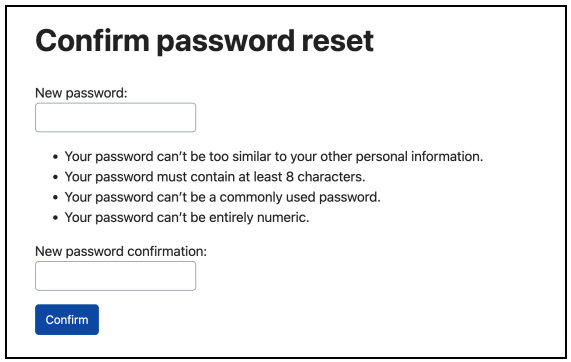
Password_reset_form.html - form to prompt user to enter email



Password_reset_form.html - after hitting “RESET” above



Password_reset_cofirm.html - after clicking on link sent in the email



Password_reset_complete.html - after updating password in the page above

> Problem: Fails silently if the email to be sent to isn’t an email of a known user.

Custom PasswordResetForm that is used by password_reset

```
   #path('password_reset/', auth_views.PasswordResetView.as_view(), name='password_reset'),
   path('password_reset/',
      auth_views.PasswordResetView.as_view(form_class=forms.MyPasswordResetForm), name='password_reset'),
```

Custom form to validate the email address:

```
from django.contrib.auth.forms import PasswordResetForm
class MyPasswordResetForm(PasswordResetForm):
   def is_valid(self):
       email = self.data["email"]
       if sum([1 for u in self.get_users(email)]) == 0:
           self.add_error(None, "Unknown email; try again")
           return False
       return super().is_valid()
```

> Replace the back-end with a real mail system

Sendgridd.com is pretty easy; no cc needed:

1) Click “signup for free” (https://signup.sendgrid.com/)
2) Add an identity
3) Create API key

Configure in settings.py:

```
# TODO: Replace with os.getenv
SENDGRID_API_KEY = 'SG.DVOIgttlRU6AxFx2…'
DEFAULT_FROM_EMAIL = 'marc.brown@alaska.edu'

EMAIL_HOST = 'smtp.sendgrid.net'
EMAIL_HOST_USER = 'apikey'
EMAIL_HOST_PASSWORD = SENDGRID_API_KEY
EMAIL_PORT = 587
EMAIL_USE_TLS = True
EMAIL_BACKEND = 'django.core.mail.backends.smtp.EmailBackend'
```

> Can bring in all the auth views at once

```
   # path('login', auth_views.LoginView.as_view(), name='login'),
   # path('logout', auth_views.LogoutView.as_view(), name='logout'),

   # path('password_change', auth_views.PasswordChangeView.as_view(), name='password_change'),
   # path('password_change/done', auth_views.PasswordChangeDoneView.as_view(), name='password_change_done'),

   # path('password_reset/', auth_views.PasswordResetView.as_view(form_class=forms.MyPasswordResetForm),
name='password_reset'),
   # path('password_reset/', auth_views.PasswordResetView.as_view(), name='password_reset'),
   # path('password_reset/done/', auth_views.PasswordResetDoneView.as_view(), name='password_reset_done'),
   # path('reset/<uidb64>/<token>/', auth_views.PasswordResetConfirmView.as_view(), name='password_reset_confirm'),
   # path('reset/done/', auth_views.PasswordResetCompleteView.as_view(), name='password_reset_complete'),

   path('password_reset/', auth_views.PasswordResetView.as_view(form_class=forms.MyPasswordResetForm),
name='password_reset'),
   path("accounts/", include("django.contrib.auth.urls")),
```

Question #1 - Why is the password_reset route still needed?

Question #2 - URLs will all have “accounts/” in front of them… why does system continue to work?
