



UNIVERSITY OF THE ASSUMPTION

City of San Fernando, Pampanga

College of Information Technology and Computer Learning Sciences

First Semester | AY 2025-2026

Name: Aaron M. Laxamana	Date: 11/19/2025
Course: BSIT	Yr & Sec. 3A

Screenshots:

The screenshot shows a code editor with a Django project structure. The current file is `registration/urls.py`. The code defines URL patterns for various views:

```
from django.urls import path
from . import views
app_name = "registration"

urlpatterns = [
    path('api/register/', views.register_user, name='register_user'),
    path('api/users/', views.list_users, name='list_user'),
    path('api/users/<int:pk>', views.user_detail, name='user_detail'),
    path('login/', views.login_view, name='login_page'),
    path('login/', views.login_view, name='login_html'),
    path('logout/', views.logout_view, name='logout_html'),
    path('users/', views.users_html, name='users_html'),
]
```



UNIVERSITY OF THE ASSUMPTION

City of San Fernando, Pampanga

College of Information Technology and Computer Learning Sciences

First Semester | AY 2025-2026

```
52
53
54 def login_view(request):
55     if request.method == "GET":
56         if request.session.get("user_id"):
57             return redirect('registration:users_html')
58         return render(request, 'registration/login.html')
59
60     email = request.POST.get('email', '').strip()
61     password = request.POST.get('password', '')
62
63     if not email or not password:
64         messages.error(request, "Enter both email and password.")
65         return render(request, 'registration/login.html', {'email': email})
66
67     try:
68         user = UserRegistration.objects.get(email=email)
69     except UserRegistration.DoesNotExist:
70         messages.error(request, "Invalid credentials.")
71         return render(request, 'registration/login.html', {'email': email})
72
73     if check_password(password, user.password):
74         request.session['user_id'] = user.id
75         request.session['user_name'] = f"{user.first_name} {user.last_name}"
76         return redirect('registration:users_html')
77
78     if user.password == password:
79         user.password = make_password(password)
80         user.save(update_fields=['password'])
81         request.session['user_id'] = user.id
82         request.session['user_name'] = f"{user.first_name} {user.last_name}"
83         return redirect('registration:users_html')
84
85     messages.error(request, "Invalid credentials.")
86     return render(request, 'registration/login.html', {'email': email})
87
88
```

```
# ----- USERS LIST -----
@login_required_view
def users_html(request):
    users = UserRegistration.objects.all().order_by('id')
    return render(request, 'registration/users_list.html', {
        'users': users,
        'current_user': request.session.get('user_name')
    })
```



UNIVERSITY OF THE ASSUMPTION

City of San Fernando, Pampanga

College of Information Technology and Computer Learning Sciences First Semester | AY 2025-2026

The screenshot shows a code editor with Python code for a login form. The code includes HTML and CSS for the form, and a POST method for handling user input. The code is part of a project named 'MyProject'.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title>Login</title>
6     <style>
7       body {
8         font-family: Arial, sans-serif;
9         display: flex;
10        align-items: center;
11        justify-content: center;
12        height: 100vh;
13        background: #f5f7fb;
14      }
15      .card {
16        background: white;
17        padding: 30px;
18        border-radius: 8px;
19        box-shadow: 0 6px 18px rgba(0, 0, 0, 0.06);
20        width: 320px;
21      }
22      .field {
23        margin-bottom: 12px;
24      }
25      label {
26        display: block;
27        font-weight: 600;
28        margin-bottom: 6px;
29      }
30      input[type="text"],
31      input[type="password"] {
32        width: 100%;
33        padding: 8px 10px;
34        border: 1px solid #ddd;
35        border-radius: 6px;
36      }
37      button {
38        width: 100px;
39        padding: 10px;
40        border: none;
41        border-radius: 6px;
42        background: #ebebeb;
43        color: white;
44        font-weight: 600;
45      }
46      .error {
47        color: #e00000;
48        margin-bottom: 10px;
49      }
50      a {
51        text-decoration: none;
52        color: #2b6cb0;
53      }
54    </style>
55  </head>
56  <body>
57    <div class="card">
58      <h2>Sign in</h2>
59      <!( if messages ) (% for msg in messages %)>
60      <div class="error">{(% msg %)}</div>
61      <% endfor %> (% endif %)>
62
63      <form method="post">
64        <!( csrf_token )>
65        <div class="field">
66          <label>Email</label>
67          <input
68            name="email"
69            type="text"
70            value="%(email|default:'')"
71            required
72          />
73        </div>
74
75        <div class="field">
76          <label>Password</label>
77          <input name="password" type="password" required />
78        </div>
79
80        <button type="submit">Login</button>
81      </form>
82    </div>
83  </body>
84</html>
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

[19-Nov-2025 08:09:35] "GET /registration/login/ HTTP/1.1" 200 0
[19-Nov-2025 08:09:36] "GET / HTTP/1.1" 200 992
[19-Nov-2025 08:09:37] "GET /registration/login/ HTTP/1.1" 200 2913
[19-Nov-2025 08:09:40] "POST /registration/login/ HTTP/1.1" 200 1979
[19-Nov-2025 08:09:50] "POST /registration/login/ HTTP/1.1" 302 0
[19-Nov-2025 08:09:59] "GET /registration/users/ HTTP/1.1" 200 2059



UNIVERSITY OF THE ASSUMPTION

City of San Fernando, Pampanga

College of Information Technology and Computer Learning Sciences

First Semester | AY 2025-2026

A screenshot of a code editor showing a Python project structure and a template file. The project structure includes files like __init__.py, settings.py, urls.py, views.py, and wsgi.py under the main directory, and registration and templates/registration sub-directories. The templates/registration/login.html file is selected and shown in the code editor. It contains HTML and CSS code for a login form, including fields for email and password, a submit button, and a link to the homepage.

```
<style>
button {
    width: 100%;
    padding: 10px;
    border: none;
    border-radius: 6px;
    background: #2b6cb0;
    color: white;
    font-weight: 600;
}
.error {
    color: #b00020;
    margin-bottom: 10px;
}
a {
    text-decoration: none;
    color: #2b6cb0;
}
</style>
</head>
<body>
<div class="card">
    <h2>Sign in</h2>
    {% if messages %} {% for msg in messages %}
        <div class="error">{{ msg }}</div>
    {% endif %} {% endif %}
    <form method="post">
        {% csrf_token %}
        <div class="field">
            <label>Email</label>
            <input
                name="email"
                type="text"
                value="{{ email|default:'' }}"
                required
            />
        </div>
        <div class="field">
            <label>Password</label>
            <input name="password" type="password" required />
        </div>
        <button type="submit">Login</button>
    </form>
    <p style="margin-top: 10px; text-align: center">
        <a href="{% url 'homepage' %}">Back to homepage</a>
    </p>
</div>
</body>
</html>
```



UNIVERSITY OF THE ASSUMPTION

City of San Fernando, Pampanga

College of Information Technology and Computer Learning Sciences First Semester | AY 2025-2026

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows the project structure under "MyProject". Key files visible include `requirements.txt`, `settings.py`, `login.html`, `homepage.html`, `users_list.html`, `views.py`, `url.py`, `__init__.py`, `admin.py`, `models.py`, `serializers.py`, `limits.py`, `utils.py`, `views.py`, `wsgi.py`, `db.sqlite3`, `migrations.py`, and `management.py`.
- Code Editor:** Displays the content of `users_list.html`. The code is an HTML template with embedded CSS and some Python logic using Jinja2 syntax (`{{ }}`). It includes a table for displaying user data.
- Terminal:** Shows the command line interface with the following logs:
 - [19-Nov-2025 10:03:36] "GET /registration/logout/ HTTP/1.1" 302 0
 - [19-Nov-2025 10:03:36] "GET / HTTP/1.1" 200 592
 - [19-Nov-2025 10:03:37] "GET /registration/login/ HTTP/1.1" 200 1923
 - [19-Nov-2025 10:03:46] "POST /registration/login/ HTTP/1.1" 200 1929
 - [19-Nov-2025 10:03:50] "POST /registration/login/ HTTP/1.1" 302 0
 - [19-Nov-2025 10:03:50] "GET /registration/users/ HTTP/1.1" 200 2029



UNIVERSITY OF THE ASSUMPTION

City of San Fernando, Pampanga

College of Information Technology and Computer Learning Sciences

First Semester | AY 2025-2026

A screenshot of a code editor showing a Django project structure and a template file. The project structure includes 'MyFrontend', 'MyProject', 'templates' (with 'homepage.html', 'login.html', and 'users_list.html'), and 'views.py'. The 'users_list.html' file contains the following HTML and CSS code:

```
2  <html>
3      <head>
4          <style>
5              ...
6          </style>
7      <table>
8          width: 100%;
9          border-collapse: collapse;
10         background: #white;
11         border-radius: 8px;
12         overflow: hidden;
13         box-shadow: 0 6px 18px rgba(0, 0, 0, 0.04);
14     </table>
15     th,
16     td {
17         padding: 12px 16px;
18         text-align: left;
19         border-bottom: 1px solid #f0f0f0;
20     }
21     th {
22         background: #eef3fb;
23         font-weight: 700;
24     }
25     tr:hover td {
26         background: #fbfdff;
27     }
28     a.button {
29         padding: 8px 12px;
30         background: #2b6cb0;
31         color: #white;
32         border-radius: 6px;
33         text-decoration: none;
34     }
35     </style>
36 </head>
37 <body>
38 <div class="top">
39     <h1 style="margin: 0">All Users</h1>
40     <div>
41         Logged in as <strong>{{ current_user }}</strong>
42         <a href="{% url 'registration:logout_html' %}" class="button" style="margin-left: 10px">Logout</a>
43     </div>
44 </div>
45 <table>
46     <thead>
47         <tr>
48             <th>#</th>
49             <th>First name</th>
50             <th>Last name</th>
51             <th>Email</th>
52             <th>Gender</th>
53             <th>Date registered</th>
54         </tr>
55     </thead>
56     <tbody>
57         {% for u in users %}
58             <tr>
59                 <td>{{ forloop.counter }}</td>
60                 <td>{{ u.first_name }}</td>
61                 <td>{{ u.last_name }}</td>
62                 <td>{{ u.email }}</td>
63                 <td>{{ u.gender|default:"N/A" }}</td>
64                 <td>{{ u.date_registered|date:"Y-m-d H:i" }}</td>
65             </tr>
66         {% empty %}
67         <tr>
68             <td colspan="6">No users yet.</td>
69         </tr>
70     {% endfor %}
71     </tbody>
72 </table>
73 </body>
74 </html>
```



UNIVERSITY OF THE ASSUMPTION

City of San Fernando, Pampanga

College of Information Technology and Computer Learning Sciences

First Semester | AY 2025-2026

A screenshot of a Microsoft Edge browser window. The address bar shows the URL "127.0.0.1:8000/registration/login/". The main content area displays a "Sign in" form. It has two input fields: "Email" containing "ddasd" and "Password" containing " ". Below the fields is a blue "Login" button. At the bottom of the form is a link "Back to homepage".
A second screenshot of the same Microsoft Edge browser window, showing the same "Sign in" form. However, the "Email" field now contains "ddasd" and the "Password" field is empty. Above the "Login" button, the text "Invalid credentials." is displayed in red. The rest of the interface is identical to the first screenshot.
A screenshot of a Windows 10 desktop. In the foreground, there is a "Snip & Sketch" window with a captured screenshot of the previous "Sign in" form. The window includes a status bar at the bottom that says "Snip saved to clipboard Select here to mark up and share the image". The desktop background shows a standard Windows wallpaper.



UNIVERSITY OF THE ASSUMPTION

City of San Fernando, Pampanga

College of Information Technology and Computer Learning Sciences First Semester | AY 2025-2026

A screenshot of a web browser window titled "Users List". The URL in the address bar is "127.0.0.1:8000/registration/users/". The page displays a table titled "All Users" with the following data:

#	First name	Last name	Email	Gender	Date registered
1	SIRFerdi	Layug	Fernand@gmail.comdasdasd	Male	2025-10-15 02:53
2	Dave	Sicat	Sicat@gmail.com	Male	2025-10-15 03:22

The browser interface includes a header with tabs for "Users List" and "New Tab", and a footer showing system status: 30°C, Partly sunny, 10:04 am, 11/19/2025.



UNIVERSITY OF THE ASSUMPTION

City of San Fernando, Pampanga

College of Information Technology and Computer Learning Sciences

First Semester | AY 2025-2026

The image shows two screenshots of a computer interface. The top screenshot is a web browser window titled "Homepage — User Registration" with the URL "127.0.0.1:8000/registration/login/?next=/registration/users/". It displays a "Sign in" form with fields for "Email" and "Password", and a "Login" button. Below the form is a link "Back to homepage". The bottom screenshot is a "Django administration" interface titled "User registrations". It shows a list of users: "AARON M. LAXAMANA CPEITEL1", "Select user registration to change", and "SIRFerdilLayug". There are buttons for "ADD USER REGISTRATION" and "WELCOME, ADMIN! VIEW SITE / CHANGE PASSWORD / LOG OUT". The taskbar at the bottom of the screen shows various open applications and system icons.



UNIVERSITY OF THE ASSUMPTION

City of San Fernando, Pampanga

College of Information Technology and Computer Learning Sciences

First Semester | AY 2025-2026

Github Link GitHub - amlaxamana/BACKENDLab3