

Unit 6 - Django and Vue.js

Classwork: The classwork for this unit should be saved in a new folder: **your_repo/unit_6/**

Homework: Homework should be saved here: **your_repo/homework/unit_6/**

*** Day 01 ***

Django and Vue.js

We learn how to develop a backend application with Django. We learn how to do the frontend like a pro with vuejs 😊

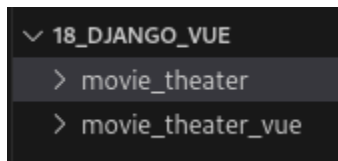
There are more features on the front-end that you can still learn, but you have learned enough to make a beautiful front-end already. If you want, you can learn more on your own, like using a router to have multiple pages in one vue application, using Pinia for state management, ...

Well, now let's make them work together. And then you can tell that you are a full-stack developer!!!

Let's see how to make them work together. You know almost everything, we just need to link them together, which is just a matter of configuration (which can be a pain with JS because for me, nodejs is a mess) and then a matter of requests between them.

Configure Django and Vue.js

1. Copy the last django application version you have (authentication lesson). Remember, you need to have communication with the database. If you need to set up part, please set it up.
2. Create a new python env for this lesson. Make sure to have **nodeenv in the dev requirements** (now, we are going to need it as you are expert on front-end).
3. Include a new package on your **main requirements: django_vite**
4. Create a new env for nodejs.
5. Let's create a new vuejs project. So what I like to do is to have a Django folder, and at the same level, the vuejs folder.



6. Check that your vue js app is working.

Insert Vue.js pages into Django templates

1. Let's create a folder apps in the src folder in the vuejs folder. In this apps folder, let's create a movie_edit folder.
2. Let's create a js file movie_edit.js inside the movie_edit folder, that is going to have the vuejs app and mount it in the html (but we still do not know which html, but that does not matter for now, it is going to be mounted somewhere).

```
import 'vite/modulepreload-polyfill';

import { createApp } from 'vue';
import App from './MovieEdit.vue'

createApp(App).mount("#app")
```

3. Let's create the file MovieEdit.vue in the same folder just to have something in the template.

```
<template>
  <div>
    This is the page where we are going to edit movie in vuejs
  </div>
</template>

<script>

export default {
  name: 'App',
  components: {
  },
  data: function() {
    return {

    }},
  methods: {

  },
}
</script>
```

4. Let's rewrite the vite.config.js. This file will define the multiple vuejs app that we can write, it will also define where we want to build our .js file for production.

```

import { defineConfig } from "vite";
import vue from "@vitejs/plugin-vue";

const backendPath = '../movie_theater';

// https://vitejs.dev/config/
export default defineConfig({
  plugins: [vue()],
  base: '/static/vite/',
  server: {
    watch: {
      ignored: [],
    },
  },
  build: {
    manifest: true,
    emptyOutDir: true,
    outDir: backendPath + '/core/static/vite/',
    rollupOptions: {
      input: {
        vue_movie_edit: './src/apps/movie_edit/movie_edit.js',
      },
    },
  },
});

```

5. What we are going to use right now, it is the rollupOptions, input:
<https://vitejs.dev/config/build-options#build-rollupoptions>,
<https://rollupjs.org/configuration-options/#input>, <https://github.com/MrBin99/django-vite>)

Let's add '**django_vite**' to the **INSTALLED_APPS** list in Django app setting.py

Also, add the following lines at the end of settings.py:

```

DJANGO_VITE_ASSETS_PATH = os.path.join(BASE_DIR, "core", "static", "vite")
DJANGO_VITE_DEV_SERVER_PORT = get_secret("vite_dev_server_port")
DJANGO_VITE_STATIC_URL_PREFIX = "vite/"
DJANGO_VITE_DEV_MODE = True # This line has to be removed in production

```

In the #Static file section in the same settings.py file add STATIC_ROOT. Otherwise, django_vite is going to complain. This is a parameter only used in production to serve the static files, through a web server like nginx, so that's why you set a directory just for that, and when you ask django to build the static files with the command "python manage.py collectstatic", it will put all the static files in there.

```

STATIC_ROOT = os.path.join(os.path.dirname(BASE_DIR), "movie_theater_static")

```

6. Do not forget to set 'vite_dev_server_port' in the secrets.json to 5173 (could be different in your computer, double check). We add this port here in case we want to have the flexibility later to change the vite server port.

```
"vite_dev_server_port": "5176"
```

7. Good, let's continue. We are going to work on a Django template now, the one to edit a movie (movie_form.html)

Right now, we have something like this:

```
{% extends "base.html" %}

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

{% endblock content %}
```

On top of the template, we are going to add:

```
{% load django_vite %}
```

And at the end of the template:

```
{% block js %}
    {{ block.super }}

    {% vite_hmr_client %}
    {% vite_asset 'src/apps/movie_edit/movie_edit.js' %}
{% endblock js %}
```

We should have something like the following code. We are going to keep the original form just in case for now.

```

{% extends "base.html" %}
{% load django_vite %}

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

<br><br>

<div id="app">
    <app></app>
</div>

{% endblock content %}

{% block js %}
    {{ block.super }}

    {% vite_hmr_client %}
    {% vite_asset 'src/apps/movie_edit/movie_edit.js' %}
{% endblock js %}

```

8. At this point, if you refresh the website page on a new movie or update a movie, you should see the part coming from vue js.

If it is working, you can even change something on the MovieEdit.vue and see the change almost immediately on the html page. (this is the magic of the vite server).

If you get it until there, that is great. We did a lot of configuration and now we are able to load a vuejs app in a Django template. That is great!!!!