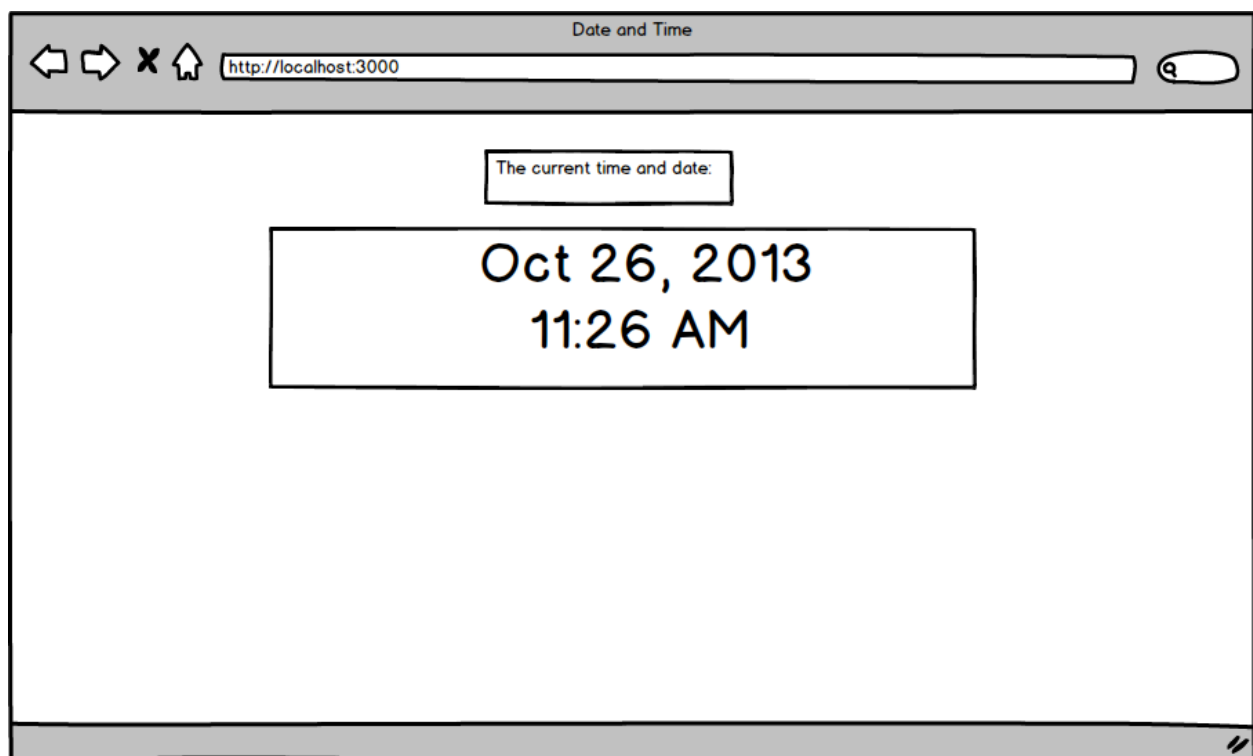


Assignment: Time Display

Objectives:

- Practice setting up a Django project
- Familiarity with passing data to a template
- Practice connecting to static files

Create a Django project with a single app called `time_display`. When you go to `localhost:8000` or `localhost:8000/time_display`, this should run a method in your controller file (`views.py`) that renders a template displaying the current date and time.



There are many ways to get the current time in Python. For example, you could have `views.py` import `gmtime`, `strftime` from `'time'` and pass the appropriate string to the render method. For example, your `views.py` might look something like this:

```
from django.shortcuts import render
from time import gmtime, strftime

def index(request):
    context = {
        "time": strftime("%Y-%m-%d %H:%M %p", gmtime())
    }
    return render(request, 'index.html', context)
```

To learn more about strftime, see

<https://docs.python.org/3.3/library/time.html?highlight=time.strftime#time.strftime>

Please also see <https://stackoverflow.com/questions/466345/converting-string-into-datetime>

Recognize that working with time - [especially timezones](#) - is among the more frustrating parts of computer programming. Do not spend more than 15 minutes exploring timezones. We will have numerous opportunities to discuss the challenges of timezones. Essentially, we want to store any timestamps in our database in UTC, and eventually use JavaScript to get the time from the user's browser to customize how times are displayed. For now, the easy fix is to set your Django settings to the timezone that works for you and move on. You have more important things to cover at this part of your career as a developer than timezones.

1. Create a new project with a single app
2. Have the root route ('/') display the current date and time
3. Incorporate your own custom stylesheet
4. NINJA BONUS: Come up with a different way to retrieve the datetime