

Procesboek Project (Alleen Opdrachten)

Wiki

Toen ik begon aan wiki, had ik zelfs na het kijken van de bijbehorende lecture eigenlijk geen idee waar ik moest beginnen. Zoveel verschillende documenten om tegelijk op te letten was overweldigend. Het hielp ontzettend om de aanpak van andere mensen die django doen te horen. Ik overlegde met Laura en Fien en Fien hielp mij ook met het begin van een stappenplan schrijven.

Het bleek al snel dat de entry pagina waarop alle verschillende informatie op weergegeven zouden worden belangrijk was. Dus het was handig om hiermee te beginnen. Hierdoor liep ik al snel tegen een probleem aan: hoe ga je om met markdown files? Om hierachter te komen gebruikte ik google. Hieruit kwamen verschillende opties om een converter te schrijven, waaronder zelfs eentje die zelfs in de urls pagina geschreven moest worden. Al deze opties werkte echter niet, waardoor ik erg gefrustreerd raakte. Gelukkig was er een werkcollega waarbij ik kon vragen hoe mijn groepsleden dit aan hadden gepakt. De meeste mensen hadden een functie hiervoor geschreven in de views file. na wat proberen leek dit mij ook de handigste optie. hierna ging alles een stuk beter.

Ook kreeg ik een handige tip in een latere conversatie met mijn werkgroep om de CS50 Discord Groep te joinen. Deze pagina is heel handig om naar antwoorden te zoeken, omdat veel mensen de vraag die je zelf hebt al gesteld hebben en de antwoorden hierop makkelijk te vinden zijn. Ook is deze pagina handig, omdat deze officieel van CS50 zelf is en spoiler dus niet toegestaan zijn. Als ik even geen inspiratie had of niet afwist van een al bestaande functie kon ik deze vaak hier vinden. Zo ook bij bijvoorbeeld bij de POST en GET functies, die toch wel vrij essentieel zijn voor dit soort projecten. Ook het gebruik van een csrf_token bleek toch wel onmisbaar en vond ik op de discord.

Wat tenslotte nog het lastigste probleem was, was dat op mijn entry pagina's de titel niet werd geprint en tekst van de body werd geprint als titel bij al mijn nieuwe entries. Hiervoor heb ik de hulp van Stephan ingeroepen. Het bleek dat de entries die meegeleverd werden door git al een titel hadden in de markdown, maar die werd bij mij apart ingevoerd door de user. De handigste methode om dit probleem op te lossen was om de al bestaande entries een titel mee te geven en om de ingevoerde titels op te roepen op de html pagina met curly braces. Ook hielp Stephan mij met een token op mijn git account generen waarmee ik vervolgens makkelijk kon pushen naar git.

Commerce

Bij commerce kon ik een deel van wat ik geleerd had bij wiki gelukkig toepassen. Hoewel er meer files waren om code in te schrijven, was het toch minder intimiderend dan aan het begin bij wiki. Ook hielp het erg om alles in kleine stapjes te doen en steeds te testen per onderdeel of het werkte voordat ik verder ging.

Om te beginnen was het wennen om te werken met models, ik wist eerst niet goed hoe de classes nu precies samenwerkten met files zoals views en admin. Maar naarmate programmeren 2 vorderde en ik makkelijker omging met python werd dit ook duidelijk. Het hielp ook dat ik een grote pauze had tussen het begin (kijken naar commerce) en het daadwerkelijke starten met zelf code schrijven pas in begin december, wanneer we al ver waren met prog2.

Nog een voorbeeld hiervan is het gebruik van de `__str__` functie waarvan ik door prog2 al wist hoe het werkte.

Wat ik vooral leerde gedurende het begin is dat het grootste deel van de logica/het programmeren in de models and views files moest. En dat vanuit hier bijvoorbeeld functies en variabelen worden aangeroepen in de html code pagina's.

Het eerste grote probleem wat ik tegenkwam was deze error waardoor de pagina niet meer wilde laden:

```
NoReverseMatch at /
Reverse for 'listing' not found. 'listing' is not a valid view function or pattern name.

Request Method: GET
Request URL: http://127.0.0.1:8000/
Django Version: 4.1.3
Exception Type: NoReverseMatch
Exception Value: Reverse for 'listing' not found. 'listing' is not a valid view function or pattern name.
Exception Location: /home/lis/local/lib/python3.8/site-packages/django/urls/resolvers.py, line 828, in _reverse_with_prefix
Raised during: auctions.views.index
Python Executable: /usr/bin/python3
Python Version: 3.8.10
Python Path: ['/mnt/c:/users/lis/Documents/Programming/project/project-commerce-django-Lis2Master',
              '/usr/lib/python38.zip',
              '/usr/lib/python3.8',
              '/usr/lib/python3.8/lib-dynload',
              /home/lis/local/lib/python3.8/site-packages',
              /usr/local/lib/python3.8/dist-packages',
              /usr/lib/python3/dist-packages']

Server time: Sun, 11 Dec 2022 22:53:52 +0000
```

Na lang zoeken bleek dat ik ergens onderweg van naam ben geswitched voor de functie listing, maar vanwege de vele verschillende documenten en functies en classes was het moeilijk op te sporen. Het is gelukt door stap voor stap vanaf het begin te kijken hoe ik het allemaal had gedaan. Hierna ben ik weer meer gaan testen of de pagina nog werkte. Het bleef echter wel een probleem om alle namen van functies en variabelen te onthouden wanneer ik niet elke dag bezig was met deze opdrachten.

Ook haalde ik een tijdje `{{}}` en `{%}` door elkaar, gelukkig leverde dit geen zoektochten door vaak testen en leerde ik snel dat `{%}` voor logica zoals voo loops en if statements is en dat `{{}}` is voor het aanroepen van variabelen uit views in html code.

Vervolgens had moeite met de cards van de listings in een rij naast elkaar te krijgen. Dit bleek uiteindelijk te komen door een `</div>` te weinig, waardoor de listings als één lang blok werden gerenderd. Een zelfde soort probleem ontstond later, toen ik comment probeerde te implementeren doordat de for loop die door de database van comments itereerde pas beëindigd werd, door de afsluitende `` van de de unordered comment list, voordat de for loop beëindigd werd. Hierdoor werden de comments helemaal niet gerenderd.

Ook stopte de gehele pagina met werken nadat ik bidding probeerde te implementeren. Dit kwam doordat ik de prijs afhankelijk had gemaakt van de class Bid. Er stonden echter al twee listings in de database die een aparte prijs hadden. Hierdoor kon ik de aanpassingen niet migraten. Ik dacht dit op te lossen door de oude listings te verwijderen en hierna te migraten. Echter kon ik door de aanpassingen in de code niet meer bij de admin pagina komen omdat hij de Bid price niet herkende, omdat deze nog niet gemigreerd was. Hierdoor zat ik dus vast in een soort error loop. Om dit op te lossen vroeg ik hulp aan Stephan.

```
self.connection.check_constraints()
File "/home/lisl/.local/lib/python3.8/site-packages/django/db/backends/sqlite3/base.py", line 264, in check_constraints
    raise IntegrityError(
django.db.utils.IntegrityError: The row in table 'auctions_listing' with primary key '1' has an invalid foreign key: auctions_listing.price_id contains a value '21' that does not have a corresponding value in auctions_bid.id.
lisl@LAPTOP-GM8PPL9F: /mnt/c/Users/lisl/Documents/Programming/project/project-commerce-django-LidlMaster$
```

```
OperationalError at /
no such column: auctions_listing.price_id

Request Method: GET
Request URL: http://127.0.0.1:8000/
Django Version: 4.1.3
Exception Type: OperationalError
Exception Value: no such column: auctions_listing.price_id
Exception Location: /home/lisl/.local/lib/python3.8/site-packages/django/db/backends/sqlite3/base.py, line 357, in execute
Raised during: auctions.views.index
Python Executable: /usr/bin/python3
Python Version: 3.8.10
Python Path: ['/mnt/c/Users/lisl/Documents/Programming/project/project-commerce-django-LidlMaster',
              '/usr/lib/python3.8.zip',
              '/usr/lib/python3.8',
              '/usr/lib/python3.8/lib-dynload',
              '/home/lisl/.local/lib/python3.8/site-packages',
              '/usr/local/lib/python3.8/dist-packages',
              '/usr/lib/python3/dist-packages']
Server time: Thu, 15 Dec 2022 21:43:22 +0000
```

Met hulp van Stephan heb ik toen de database verwijderd, waarna de hele website het niet meer deed. Dit was schrikke, maar na wat proberen lukte het om te migreren wanneer bij makemigrations specifiek auctions werd aangeroepen en er geen database aanwezig was. Hierna was het een simpele kwestie van opnieuw een superuser aanmaken en nieuwe listings inputten.

Het enige wat nu nog mis ging was dat de class Bid niet wist wat het met argument user moest doen, dit heb ik met hulp van Martijn opgelost. Het bleek dat in de class de user variabele buyer had, waardoor het programma in de war raakte. In plaats van user mee te geven in views bij price en bids, moest dus buyer worden gebruikt.

Tenslotte heb ik ook nog een mooie functie datetime gevonden, waarmee je kan implementeren dat posts ook vermelden wanneer deze geplaatst zijn. Dit zou heel mooi zijn bij de item listings, bids and comments. Het is wel al functioneel voor de admin/superuser, alleen moet ik het nog in zien te pluggen op de listingpagina zodat de user deze ook kan gebruiken.

Deze implementatie is ook gelukt door hem in comments gewoon aan te roepen met curly braces als comments.date. En bij listing en bids is het gelukt door hem mee te renderen en door hem mee te geven bij de functie in views met wat we eerst dachten als indexeren in een dictionary met listingData["date"], maar bleek later toch te moeten met listingData.date. Hierna kon op de HTML pagina de datum gewoon worden aangeroepen met curly braces.