# FANGOUS DAMIEN – Stellium Exam | Fixing Process

Before trying to start the application, I first went through how it was designed.

I identified the front-end served in the Client App folder and the other elements of the back-end (controllers, services, config files and startup actions).

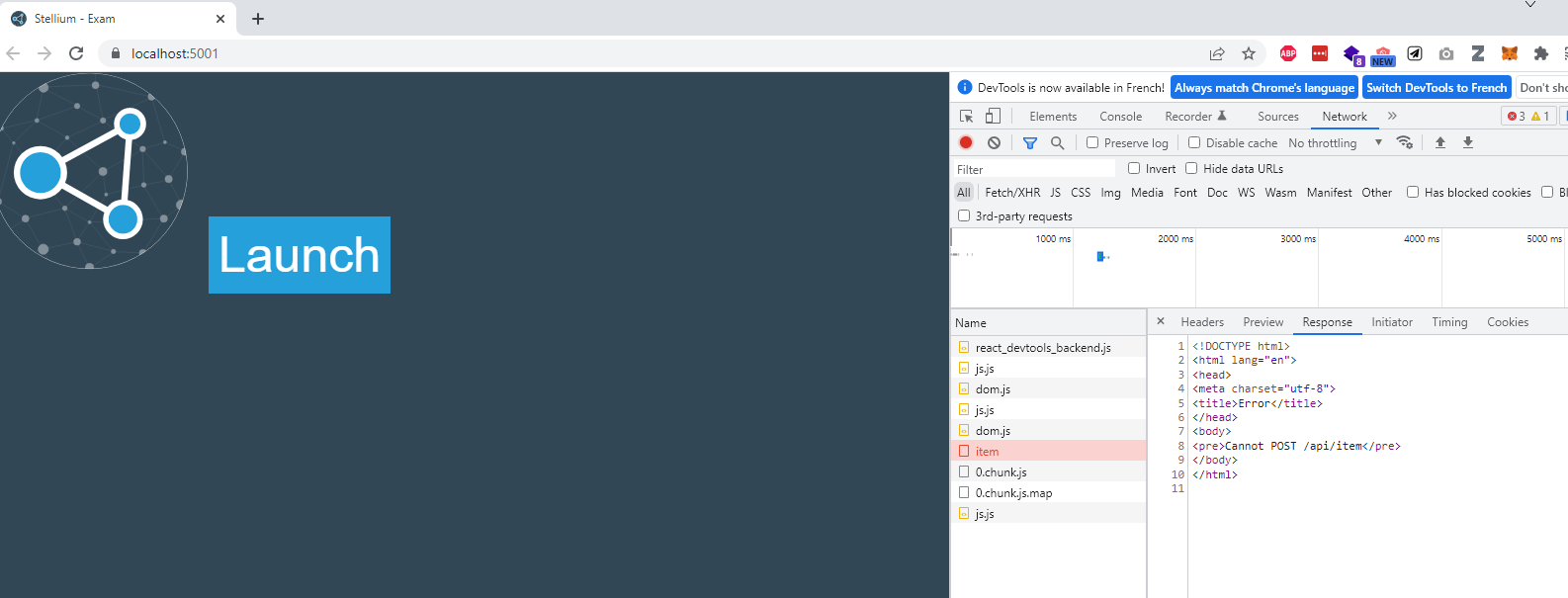
I first installed node packages on the front-end, the node version I used at first was the 16 lts which didn’t work with the module node-sass version 4.14.1 (verified from the error message). Checking the documentation (<https://github.com/sass/node-sass>) I saw that the required node version was 14, I then switched to the 14 lts through nvm and the modules installation went through.

Trying to start the front, I got an error message saying that axios was missing, I installed it with “npm install axios”, recompiled the front-end and it build successfully.

Next step was to start Cosmos DB emulator, get the credentials and add them to the .Net app config file (Account and Key properties). I then tried to launch the app, it built and started. I checked the DB with the data explorer to see if the connection worked and the container had been created, it did.

Connecting to the app’s url gave me a proxy error, the app could not connect to the front. I check how the forwarding was managed, in this case it could have been using the react development server (spa.UseReactDevelopmentServer(npmScript: "start");) or forwarding to a port. Looking into the Startup.cs file, I saw that it was forwarding on port 3001, I started the front-end separately and noticed that it was being served on port 3000. I modified the forwarding port and the front-end appeared.

Next I tried to click the launch button but nothing happened, I checked the console and saw that there was an error, I switched to the network tab to check the request and saw that I received a error page with “Cannot POST…” :

We don’t usually make a POST request to get stuff, I checked the route on the back-end and it’s indeed a GET action. I check the front-end service and saw that it was doing a post, I fixed it but still got the same html error but now with a “Cannot GET…”.

I added a Swagger with the package Swashbuckle to test my route more easily.

I added a breakpoint in the ItemController to check when it was called. I started the app, went to the swagger page and called the endpoint, I didn’t hit the breakpoint. My controller was not called, I checked the controller attributes, it looked good. I kept going up, the controller were registered in the services collection (services.AddControllersWithView()). I noticed that the endpoint were not mapped, I added the mapping, tried to call the endpoint and finally hit the breakpoint but the method returned nothing.

The Startup had a TODO to add the item to the container. I wrote the bit of code to add the item with the content that was in the config file and started the front-end and back-end again. I got a new error in the front-end concerning an undefined variable in the “atob” method in the Home file. I checked it and noticed that it was trying to access the property content of the GetStuff() result. The Item returned from the back-end had only an id and a description property, no content. I renamed content as description in the Item model in the front-end and changed the property accessed in the Home.tsx to match it.

The application worked and I got the result page to display but did not looked like the expected result. I added some css in the Home.module.scss to match how it was expected to looked like. And done.