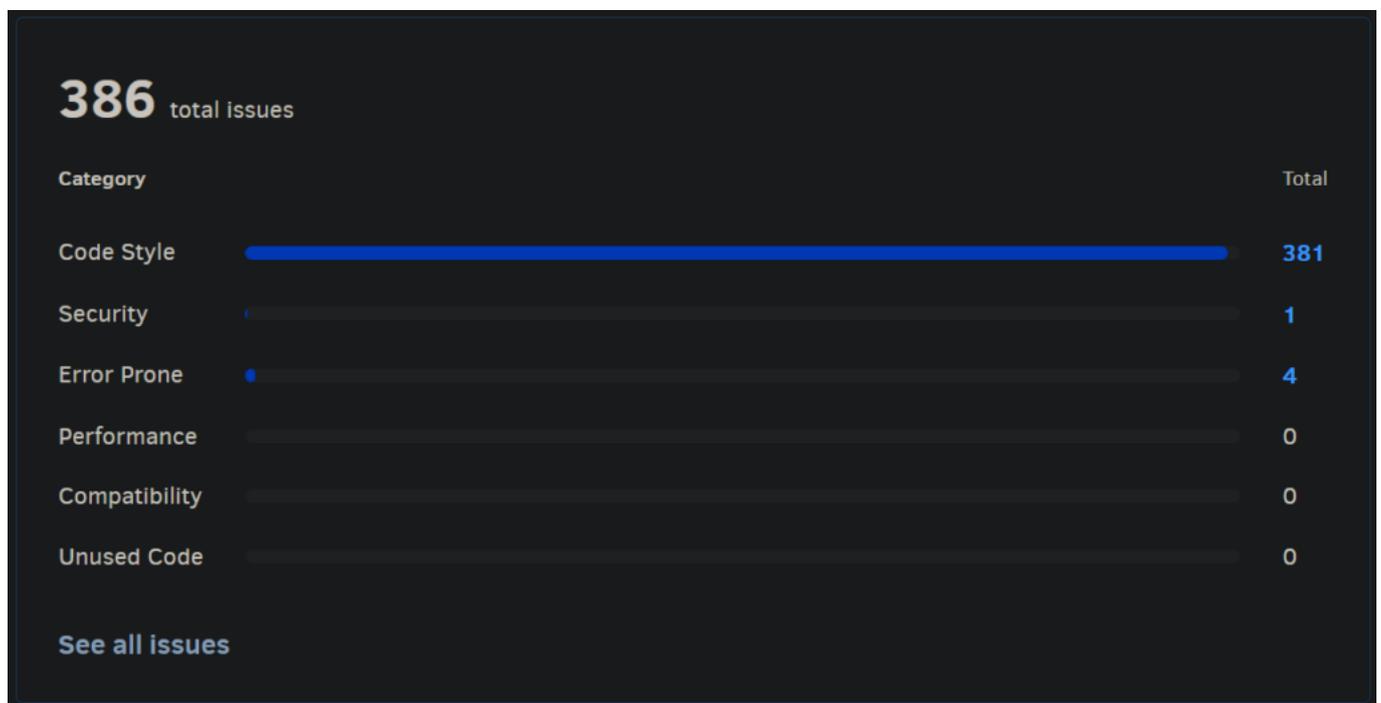


Hello again!

This week, we had a look at the code metrics from codacy. As we already integrated codacy in our workflow, it makes sense to look at the change proposals so we also have a more stable and more secure application.

The following screenshots show you the issue dashboard as well as a detailed description of the Security and Error Prone issues:



Metrics Breakdown before refactoring

```
Crawler/api.py
Possible binding to all interfaces.
35 app.run(host='0.0.0.0', port=22026)
```

Security issues reported by Codacy

```
Crawler/Dockerfile
Always tag the version of an image explicitly
2 FROM python

Use COPY instead of ADD for files and folders
7 ADD ./crawler

Frontend/src/app/components/hot-deals-widget/hot-deals-widget.component.css
Expected RBRACE at line 6, col 38.
6 grid-template-columns: repeat(2, 1fr);

Expected RBRACE at line 15, col 35.
17 grid-template-rows: repeat(5, 1fr);
```

Error Prone issues reported by Codacy

Security Issues can possibly make our application vulnerable to malicious users and Error Prone issues could cause in the application not working properly.

The security issue says that it's not good to bind an application to all interfaces (That's what happens when you just say 0.0.0.0 and also don't specify any protocol). But this is basically a non-issue as the crawler runs in a docker container and hence behind a reverse proxy that already filters and routes all the HTTP traffic. This is why we are not going to adress this issue.

Two of the Error Prone issues complain about our docker file. We fixed both of them as they ensure that every re-deployment of the crawler uses the right version of python and all the required files are copied correctly to the docker container.

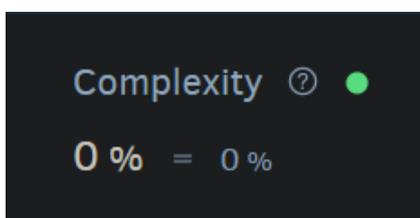
The other two Error Prone issues complain about wrong CSS Syntax which actually isn't the case, so we think there might be some problem with codacy or maybe we just use another version of CSS.

The following screenshot shows the dashboard after our fix:



Here you can find our [Codacy Dashboard](#) and our [Fix-Commit](#).

We also had a look at the code complexity codacy told us, but as you can see in the following screenshot, the complexity is 0% for us, so no issues here:



Codacy Code Complexity

This is one of our last blog posts, so thanks already for sticking around. We hope that you also follow along until our very last post.

Until then, have a great week.

Team Betterzon