Documentation detection plugin

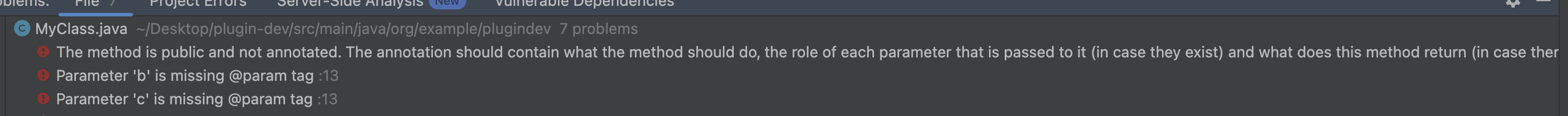
Lorena Grigorescu 248-1

DocDev is a java plugin that is built to detect all public methods that are not fully documented. It can be found in IntelliJ under the plugins section, more precisely in the marketplace.

A screenshot of a computer

Description automatically generated

The purpose of this plugin is to detect all public methods in java that are fully documented. If there is any method that is not properly documented, then IntelliJ reports this as an error. It analyzes all methods in all classes where the plugin is installed.



If the method is not documented at all, a message like this will appear "The method is public and not annotated. The annotation must contain what the method should do, the role of each parameter passed to it (if any) and what this method returns (if there is no void function);", and if it is documented, but partially, a message will appear mentioning all the arguments that have not been documented.

To build this plugin, we used PSI (Program Structure Interface) which is an API that facilitates developers to analyze the textual content of written code. With the help of predefined objects (PsiMethod, PsiTag etc.) we were able to extract each public method that we analyzed and for each of the existing public methods, we searched through the entire parameter list and checked if in the documentation each parameter is preceded by "@param".

A computer screen with white text

Description automatically generated

A computer screen shot of a code

Description automatically generated

Another thing worth mentioning is that the class that handles the implementation of the plugin itself must implement a special class called Annotator and implicitly the

**@Override public void annotate(@NotNull PsiElement element, @NotNull AnnotationHolder holder)** method**.**

As build automation tools I used gradle.

In order to be visible in the marketplace, I configured a special .xml file (plugin.xml) in which I specified its name (DocDev), what its purpose is (between the <description> tags), who its author is and what dependencies it uses.

A screenshot of a computer program

Description automatically generated