Homework_5

Selecting an OSS License

The goal of this assignment is to help you understand different types of open source licenses and practice using online tools to evaluate and select the most suitable license for a given software project.

Scenario:

You are working on a **collaborative open source project: "Prose-Builder"**, a simple tool to generate prose based on sentences imported from *dependencies*. You and your team want to **publish** this project **on GitHub** but are unsure **which *open source license* best suits** your goals.

You want:

• Others to **use and share the code freely**.
• **Modifications** to be **allowed**, but you want **credit**.
• The modified **versions should stay open source**.
• **No commercial restrictions**.

Steps:

1. Compare at least 3 licenses that could be potential candidates for your project.
2. Use all three tools listed above to analyze their key features, obligations, and restrictions.
3. Choose one license that you believe best fits the ~~EduNotes~~ project goals.
4. Justify your choice in 300–400 words.

Include:

- Key features of the selected license.
- How it meets the project's objectives.
- Why it is a better fit than the other licenses you compared.
- Mention the source you got your facts from.

Exercise 1: License Selection

To determine which software license would be the most appropiate for the criteria defined above, I started with the OSS Watch License Differ Tool:
The amount of options to choose from was a little overwhelming at first. However, I found this tool to be the most effective. Here are the results of my search (I chose strong and weak copyleft for comparison, assuming copyleft would be the only way to guarantee the code stays open source in any case):

> **Summary**
> This is your selection:
>
> 1. Popular and widely used: -
> 2. Licence type: Strong/weak copyleft (Img 1/2)
> 3. Jurisdiction: -
>   4.a Grants patent rights: -
>   4.b Patent retaliation clause: -
> 4. Specifies enhanced attribution: -
> 5. Addresses privacy loophole: -
> 6. Includes 'no promotion' feature: -

1 Strong copyleft

Affero GNU Public License	[2 out of 2]
European Union Public License	[2 out of 2]
GNU General Public License	[2 out of 2]
GNU General Public License v3.0	[2 out of 2]
IPA Font License	[2 out of 2]
Non-Profit Open Software License 3.0	[2 out of 2]
Open Software License 3.0	[2 out of 2]
Reciprocal Public License 1.5	[2 out of 2]
Simple Public License 2.0	[2 out of 2]

2 Weak copyleft

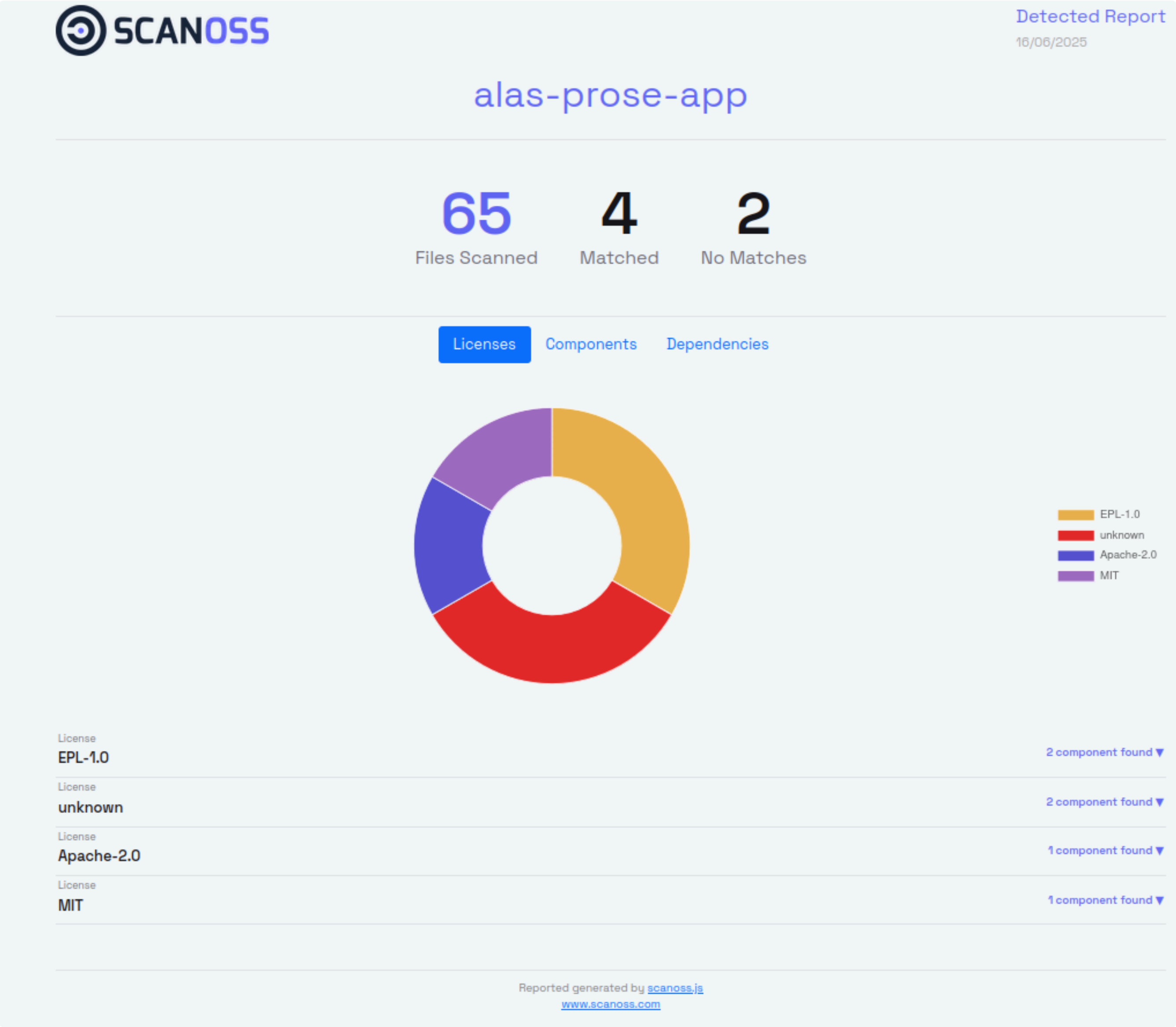
Adaptive Public License	[2 out of 2]
Common Development and Distribution License	[2 out of 2]
Common Public Attribution License 1.0	[2 out of 2]
Common Public License 1.0	[2 out of 2]
Eclipse Public License	[2 out of 2]
GNU Library or 'Lesser' General Public License	[2 out of 2]
GNU Library or 'Lesser' General Public License v3.0	[2 out of 2]
Microsoft Reciprocal License	[2 out of 2]
Mozilla Public License 1.1	[2 out of 2]

I decided to focus on the European Union Public and GNU (L)GP Licenses from here.

However, as one result of my own research for more information on OSS licenses was
5 types of software licenses you need to understand[1]

> **How to tell what licenses apply to a particular software codebase**
> Before you can determine which licenses govern any components in a codebase, you need a software Bill of Materials (SBOM), a list of all the components in the code.

I installed *ScanOSS SBOM Workbench* through the Linux Mint Software Manager and got the following results:



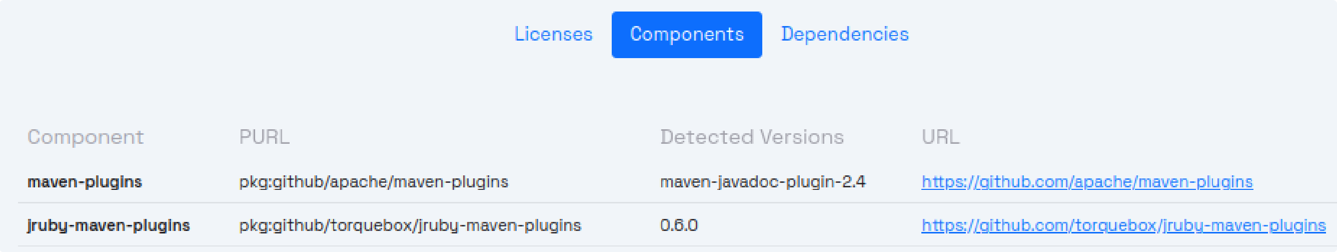
Detected Report
16/06/2025

alas-prose-app

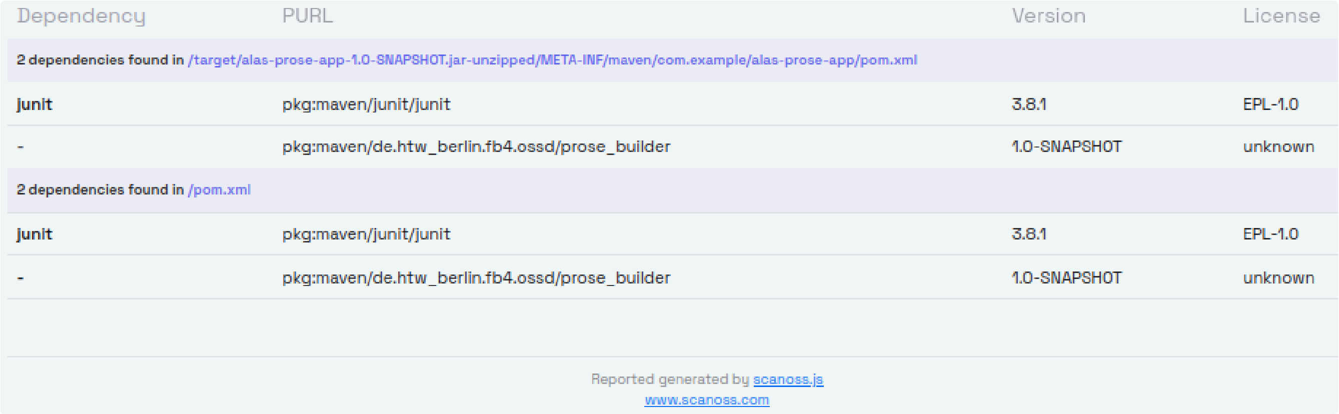
65	4	2
Files Scanned	Matched	No Matches

License	
EPL-1.0	2 component found ▼
unknown	2 component found ▼
Apache-2.0	1 component found ▼
MIT	1 component found ▼

Reported generated by scanoss.js
www.scanoss.com



Component	PURL	Detected Versions	URL
maven-plugins	pkg:github/apache/maven-plugins	maven-javadoc-plugin-2.4	https://github.com/apache/maven-plugins
jruby-maven-plugins	pkg:github/torquebox/jruby-maven-plugins	0.6.0	https://github.com/torquebox/jruby-maven-plugins



Dependency	PURL	Version	License
2 dependencies found in /target/alas-prose-app-1.0-SNAPSHOT.jar-unzipped/META-INF/maven/com.example/alas-prose-app/pom.xml			
junit	pkg:maven/junit/junit	3.8.1	EPL-1.0
-	pkg:maven/de.htw_berlin.fb4.ossd/prose_builder	1.0-SNAPSHOT	unknown
2 dependencies found in /pom.xml			
junit	pkg:maven/junit/junit	3.8.1	EPL-1.0
-	pkg:maven/de.htw_berlin.fb4.ossd/prose_builder	1.0-SNAPSHOT	unknown

Reported generated by scanoss.js
www.scanoss.com

Since neither MIT nor Apache 2.0 meet the "modified versions must be open source" criterion, I took a closer look at the EPL-1.0 and also decided to compare GNU GPL v3.0 to all of the above, because it is quite popular.

Comparison of MIT, Apache 2.0, EPL-1.0, EUPL, and GNU GPL

License	Use/Share Freely	Modifications Allowed (Credit)	Modified Versions Must Stay Open Source	Commercial Use
MIT	Yes	Yes, with credit	No	Yes
Apache 2.0	Yes	Yes, with credit	No	Yes
EPL-1.0	Yes	Yes, with credit	Yes (file-based copyleft)	Yes
EUPL	Yes	Yes, with credit	Yes (strong copyleft)	Yes
GNU GPL	Yes	Yes, with credit	Yes (strong copyleft)	Yes

Because of the nature of the project, I decided to emphasize strong copyleft. The purpose of prose builder is rather educational or even recreational. In my opinion there is no need to consider future use of its code in proprietary projects, and the risk of users shying away is negligible.

This left the EUPL and GNU GPL as options.

Here, I was indecisive:

GNU GPL is more common, by far. Potential users would probably feel safer with a license they encountered many times before. However, any derivatives must be under GNU GPL as well, which might cause compatibility problems.

EUPL, on the other hand, is not only specifically designed for use in the EU. It is also translated in most, if not all EU languages. Therefore, legal concerns are easier to understand – within the EU, where I consider the software to be of the most interest. In terms of license interoperability, it is much more flexible (see https://interoperable-europe.ec.europa.eu/collection/eupl/matrix-eupl-compatible-open-source-licences#section-2).

With recent events in mind, I finally opted for the European Public License. I think it is crucial that IT-related topics start to be handled outside US (centered) legislation, regulation and licensing. The need for free and open source software will increase in face of global problems such as climate change, political and societal challenges. Not least, true free and open source software provides the chance for people to learn about technologies that surround them and get involved.

Conceptual Questions

Answer the following short-answer questions (100–150 words each):

5. What is the difference between permissive and copyleft open source licenses? Provide one example of each.

> **Permissive**
> Use, modification and redistribution are allowed with minimal restrictions. Modifications/derivatives are not required to use the same license or stay open source. Usually, original copyright notice and license text are sufficient to retain. Example: MIT License
> **Copyleft**
> To ensure modifications stay open source and available to the public, any modified or distributed derivative code must be licensed under the original license. When used in proprietary software, the source must be disclosed.
> Example: GNU GPL

6. Why is it important to choose a license when releasing open source software?

> Surprisingly, if no license is chosen the code is protected by copyright law, resulting in others not being able to legally use, modify or share it.
> In order to grant and encourage collaboration and adoption, an open source license is required. This also provides legal protection for one self as clarity for users and fellow developers. permissions and restrictions apply accordingly to the chosen license, which might also reflect your values and/or the project's goals.

7. Who grants you use rights for the Linux Kernel, if you buy a copy of Red Hat Enterprise Linux? Justify your answer.

> Since the kernel itself is published under GNU GPL v2 since version 0.12, the use is granted by its authors.[2]
> Red Hat Enterprise Linux (RHEL) as a company provide business level support and solutions for various use cases (edge/high performance computing, SAP support, cloud and SQL server deployments)[3]
> They use the Linux kernel but this use is non-discretionary under GNU GPL.

Submission Instructions
• Submit a single PDF file that includes:
o Your license comparison table or summary.
o Your license choice and justification.
o Answers to the three conceptual questions.

---

[1] How to tell what licenses apply to a particular software codebase
Before you can determine which licenses govern any components in a codebase, you need a software Bill of Materials (SBOM), a list of all the components in the code.
[https://www.blackduck.com/blog/5-types-of-software-licenses-you-need-to-understand.html] ↩

[2] https://de.wikipedia.org/wiki/Linux_(Kernel) ↩

[3] ↩