Software Licenses

- **What are they for?**

A software license is a document that provides legally binding guidelines for the use and distribution of software.

There are many types of software licenses, a software license includes rules and requirements about how the users can utilize said software. It also establishes how fees for using the software are determined and how much the user is allowed to modify, copy, or disperse the software.

There are different types of licenses such as Public Domain License, Lesser General Public License, Permissive, Copyleft, Proprietary. The following are some open source licenses.

- **GPL (GNU General Public License - Copyleft)**

The GNU GPL is explicit about the patent grant an author would be giving when the code (or derivative work) is distributed, (the MIT license does not discuss patents). Moreover, the GPL license impacts derivative works, preserves copyright terms and is suitable for commercial, patent, and private use.

Any software that uses GPL code must distribute all its source code under the same license. So if you use GPL code in your software (e.g., by using a GPL library), and distribute your application, all your source code must be made available under the same GPL license. This restriction makes the GPL a strong copyleft license.

- **MIT (Permissive license)**
  MIT is free to do whatever we want with the licensed code or product.

  - It permits the reuse of software in proprietary projects, provided any future licensed software contains a copy of the license terms and copyright notice. The MIT license is short, clear, and easy to understand.
  - Commercializing software released under an MIT license is more challenging. A common business model involves licensing the software as open-source and offering proprietary extensions built on top of that software.
  - The only condition required to use the software is to include the same copyright notice in all copies or any substantial portions of the software.
  - It doesn't explicitly include a patent license grant. This means anyone who copies, uses, or distributes the software might become liable for patent infringement if the creator or contributor patented certain components
  - Some projects using MIT: React Native, Node js, Ruby on Rails.

- **Apache (Permissive license)**
  The Apache 2.0 license is a permissive license that is somewhat similar to the MIT license. The main difference is that the Apache license includes more specific rules governing its use and any derivatives. The Apache license is much more challenging to read for the average person because it consists of a lot of legal jargon, and it's also much longer than the MIT license's three paragraphs.

  - Apache 2.0 allows code to be reused and modified by developers, including for monetary gain.
  - The Apache 2.0 license requires that any significant modification of code under this license must be described. The actual modified code doesn't have to be released, but the changes must be described in a notification included in any released derivations.

- A benefit that Apache 2.0 licenses have over MIT licenses is that developers can claim patents on derivative projects. The language allowing patent right transference clearly specifies the terms that some users and lawyers appreciate because the degree of legal protection provided minimizes the risk of potential lawsuits.
- A limitation of this license for developers is that it requires you to add prominent notifications of any changes you make to a file.
- This is the license Google uses for most of its open-source projects

References

https://ghinda.com/blog/opensource/2020/open-source-licenses-apache-mit-bsd.html

https://soos.io/apache-vs-mit-license/

https://www.exygy.com/blog/which-license-should-i-use-mit-vs-apache-vs-gpl