People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

Mohamed Lamine Debaghine University – Setif 2
Faculty of Arts and Languages
Department of English

Course Title: ***FOSS***
Level: ***Master 1 – English Language***

Chapter I

General Introduction

to Free and Open Source Software

2025/2026

Chapter I: General Introduction

In this chapter, students will:

- **Definitions and History of Free and Open-Source Software:** Understanding what FOSS is, its origins, and its evolution.
- **Differences between Free Software, Open Source, and Proprietary Software:** Key distinctions, advantages, and limitations.
- **Philosophy and Ethical Issues of Free Software:** Principles behind software freedom and ethical considerations.
- **Types of Licenses:** Overview of common FOSS licenses, including **GPL, LGPL, MIT, Apache, BSD, and Creative Commons**.

Introduction:

The rapid development of information and communication technologies has transformed the production and use of software. In this context, **Free and Open Source Software (FOSS)** has emerged as an alternative to proprietary software, promoting **freedom, collaboration, and transparency** by allowing users to study, modify, and redistribute software.

Beyond its technical dimension, FOSS represents a **philosophical and ethical approach** that emphasizes user autonomy and shared knowledge. It has significantly influenced modern computing, Internet infrastructure, education, and research.

This chapter introduces the fundamental concepts of FOSS, outlines its historical development, and explains the differences between free software, open-source software, and proprietary software. It also presents the ethical foundations of free software and an overview of the most common FOSS licenses.

1. Definitions and history of free and open-source software:

1.1 Definition of Terms:

Free and Open Source Software (FOSS): refers to programs whose licenses allow users to:

- Use the program for any reason.
- Examine and adjust the program.
- Redistribute copies of the original or altered application without giving the prior developers' royalties.

FOSS has expanded from being largely unknown to becoming a worldwide phenomenon in recent years. However, its actual significance and ramifications are still not well understood.

1.2 history of free and open-source software:

Timeline of Free and Open-Source Software (FOSS)

Period	Key Events & Projects	Notes / Examples
Up to late 1960s	Early software freely shared	IBM, SHARE, DECUS – software was an add-on to hardware
Late 1960s – 1970s	IBM unbundles software; rise of proprietary software	Companies restrict sharing; market for software emerges
1973–1978	Early free software examples	SPICE (IC simulation), TeX (typesetting)
1972–late 1970s	Unix development & academic distribution	CSRG at UC Berkeley forms community, shares and improves code
1980s	GNU Project & Free Software Foundation	Richard Stallman: four freedoms; GCC, Emacs; FOSS philosophy established
1980s	BSD Unix, X Window System, Internet protocols	Communities collaborate; FOSS principles applied; MIT license (X Window)
Early 1990s	Linux kernel (1991), BSD & Linux distributions	Linux 1.0 (1994); NetBSD, FreeBSD, OpenBSD; first complete FOSS OSs
Mid–late 1990s	Growth of communities and companies	Debian (1993), Apache (1993), KDE (1996), GNOME (1997), Mozilla (Netscape FOSS, 1998)
2000s	Foundations & corporate involvement	Linux Foundation, Eclipse Foundation; companies fund, hire developers, or form nonprofits
Today	Global phenomenon	FOSS is widely used in software, Internet infrastructure, education, and industry

2. Differences between free software, open source, and proprietary software:

Understanding the differences between free software, open-source software, and proprietary software is essential for grasping how software is developed,

distributed, and used. Each of these models is based on distinct principles regarding user rights, access to source code, and licensing conditions. This

section highlights the main characteristics of each type of software in order to clarify their technical, philosophical, and commercial differences.

2.1Free-Software:

Free software refers to software that guarantees users four fundamental freedoms to run the program for any purpose, to study and modify its source code, and to redistribute original or modified versions. It is primarily based on **ethical and philosophical principles** that emphasize user freedom and control.

2.2 Open Source Software (OSS):

Open-source software is software whose source code is publicly available and distributed under licenses approved by the Open Source Initiative. It allows users to study, modify, and redistribute the software, with a primary focus on practical and technical benefits such as collaboration and software quality.

2.3 Proprietary Software:

Proprietary software is software whose source code is closed and owned by an individual or company. Users are granted limited rights under a commercial license, and modification or redistribution is generally prohibited.

2.4 Comparison between Free Software, Open Source Software, and Proprietary Software

Aspect	Free Software	Open Source Software	Proprietary Software
Source code	Open	Open	Closed
User freedoms	Central	Allowed	Restricted
Main focus	Ethical/philosophical	Practical/technical	Commercial
Modification	Allowed	Allowed	Not allowed

Redistribution	Allowed	Allowed	Restricted
Development model	Community-based	Collaborative	Closed team

3. Philosophy and ethical issues of free software.

- The principle of free software is based on granting users the right to control the software they use.
- It emphasizes freedom rather than price, meaning that "free" refers to liberty, not cost.
- The philosophy of free software promotes the four essential freedoms: running programs, studying them, modifying them, and redistributing them.
- Free software supports transparency, allowing users to examine the source code and understand how programs function.
- It encourages sharing and collaboration, viewing knowledge as a public good rather than private property.
- From an ethical perspective, restricting access to source code is considered unfair and harmful to users.
- Free software helps prevent vendor lock-in, ensuring that users are not dependent on a single company.
- It promotes social responsibility, equal access, and respect for users rights.

4. Types of licenses

Software licenses play a crucial role in defining how software can be used, modified, and redistributed. In the context of Free and Open Source Software, licenses are designed to protect users' freedoms while establishing legal frameworks for collaboration and distribution. This section presents an overview of the most common licenses used in the FOSS ecosystem and highlights their main characteristics.

❖ **GNU General Public License (GPL)**

The GPL is a copyleft license that guarantees users the freedom to use, study, modify, and redistribute software. Any modified or derived work must be released under the same license, ensuring that the software remains free.

❖ **GNU Lesser General Public License (LGPL)**

The **LGPL** is a weaker version of the GPL, mainly designed for **software libraries**. It allows proprietary software to link to the licensed library while preserving freedom for the library itself.

❖ **MIT License**

The **MIT License** is a **permissive license** that allows reuse, modification, and distribution of software, including in proprietary projects, provided that the original copyright notice is preserved.

❖ Apache License (Version 2.0)

The **Apache License** is a permissive license that allows commercial use, modification, and distribution. It also provides **explicit patent protection** for users.

❖ BSD License

The **BSD License** is a permissive license with minimal restrictions. It allows redistribution and use in both open-source and proprietary software, requiring only attribution to the original authors.

❖ Creative Commons Licenses (CC)

Creative Commons licenses are primarily used for **non-software content**, such as documentation, educational materials, and media. They allow creators to specify how their work can be reused.

❖ Comparison of License Types

License	Type	Modification	Redistribution	Proprietary Use
GPL	Copyleft	Allowed	Allowed (same license)	Not allowed

LGPL	Weak copyleft	Allowed	Allowed	Limited
MIT	Permissive	Allowed	Allowed	Allowed
Apache	Permissive	Allowed	Allowed	Allowed
BSD	Permissive	Allowed	Allowed	Allowed
Creative Commons	Content licenses	Allowed (varies)	Allowed (varies)	Varies

Conclusion

This chapter presented the core concepts of Free and Open Source Software, including its definitions, historical development, philosophical principles, and licensing models. It showed that FOSS is more than a technical alternative, as it is grounded in values of freedom, collaboration, and transparency.

The comparison between free software, open-source software, and proprietary software highlighted their differences in terms of user rights, development models, and objectives. In addition, the overview of major licenses illustrated how legal frameworks protect software freedoms and regulate its use and distribution.

Overall, this chapter provides a solid foundation for understanding FOSS and prepares students for more advanced topics related to its applications and impact in the following chapters.

References

[1] Gonzalez-Barahona, J. M. (2021). *A Brief History of Free, Open Source Software and Its Communities*. **IEEE Computer**, Open Source Expanded, February 2021.

[2] Stallman, R. M. (2002). *Free Software, Free Society: Selected Essays*. Boston: GNU Press.

[3] Free Software Foundation (FSF). *GNU General Public License (GPL)*. Source: Free Software Foundation.

[4] Free Software Foundation (FSF). *GNU Lesser General Public License (LGPL)*. Source: Free Software Foundation.

[5] Massachusetts Institute of Technology (MIT). *MIT License*. Source: MIT.

[6] Apache Software Foundation. *Apache License, Version 2.0*. Source: Apache Software Foundation.

[7] University of California, Berkeley. *BSD License*. Source: University of California, Berkeley.