linfo2401

2022

Open Source strategy for software development

5.00 credits	30.0 h + 15.0 h	Q1

Teacher(s)	Dricot Lionel ;
Language :	English > French-friendly
Place of the course	Louvain-la-Neuve
Prerequisites	Corequis : LINGI2255
Main themes	Introducing the Open Source approach comparing it systematically to the owner approach in the context of choosing an application; analyze interest to contribute to open source software development, if this approach is selected. Analysis and implementation of an Open Source approach for the development of an application; opportunity to distribue a solution in an Open Source application, eg with a valorization or image target. Open Source approach for internal developments • Objectives of an open source approach • Advantages and difficulties of this approach • Practical implementation Development of Open Source Products • Objectives of an open source approach • Integration and management of an Open Source community • Copyright and open source license choice • Valuation Method and economic model • Practical implementation
Learning outcomes	**At the end of this learning unit, the student is able to :** Given the learning outcomes of the "Master in Computer Science and Engineering" program, this course contributes to the development, acquisition and evaluation of the following learning outcomes: • INFO1.2 • INFO6 Given the learning outcomes of the "Master [120] in Computer Science" program, this course contributes to the development, acquisition and evaluation of the following learning outcomes: • SINF6 Students completing successfully this course will be able to : 1 • Understand and explain the advantages and disadvantages of Open Source approach of internal software development • Formulate a position between opensource and owner approach for the development of a given application in an organization • Justify the choice of an open source approach of software development within an organization • Justify the choice of an owner approach compared to an open source one; • Organize a development according to an open source approach • Plan the progress to reach the objectives in a coherent way • Understand and explain the advantages, disadvantages and economic models related to a distribution of open source software • Justify the choice of a distribution of open source software developed • Organize a collaborative Open Source development • Justify the choice of a contributory approach in software development
Evaluation methods	Oral exam during which the student will defend his project (50% of the points) and answer the teacher's questions (50% of the points) The modalities remain the same in September (the project being individual and can be completed in July/August).
Teaching methods	Ex-cathedra course, with external interventions and individual project to be implemented during the year. The deadline for submitting the project is two weeks before the exam or end of the semester.

Content	This course proposes to answer these questions through a theoretical course illustrated by real and documented examples as well as through a practical approach encouraging students to contribute to open source. The theoretical approach will address the following areas: • History of Open Source, Free Software, GNU and major open source projects. • Legal aspects of open source: licenses. • Political aspects of the use of open source software. Example of the city of Munich and the Limux project. • Aspects of community governance: how to contribute, manage and support an open source project? How to manage a community and its conflicts? Who decides on the code, the releases, the documentation? Examples of the GNOME project and the Linux kernel. Open source communication tools. • Economic aspects: the different open source business models. successes and failures. Historical examples from Red Hat, Ubuntu and Mandriva. Exploring new models with OpenCollective • Technical and political aspects of open source 1: security, respect for privacy. • Technical-political aspects of open source 2: interoperability and open formats. • Open source in the age of the web: the challenges, the AGPL license. Example of React and AngularJS projects • Open Source and decentralization. Examples of the XMPP, Diaspora, Mastodon and Bitcoin projects. Introduction to blockchain. • Case Study 1: Recommend an open source solution within an enterprise. Concept of Total Cost of Ownership. • Case study 2: Using open source in cooperation with proprietary software: the different scenarios. Example of Android. • Case study 3: open source an existing proprietary application. Example from Mozilla. As practical work, students will have to produce throughout the course a report analyzing an open source project of their choice. In addition to this analysis, they will be invited to make a contribution to the project, even simple or non-technical, and to describe the process of this contribution.
Faculty or entity in charge	INFO

Programmes containing this learning unit (UE)

Program title	Acronym	Credits	Prerequisite	Learning outcomes
Master [120] in Computer Science and Engineering	INFO2M	5		🔍
Master [120] in Computer Science	SINF2M	5		🔍
Master [60] in Computer Science	SINF2M1	5		🔍