OPEN_NEXT

Deliverable 2.6

Final release of models and standards for design reuse:

Models and standards for documentation and knowledge capture enabling reuse of OSH design

This project is funded by the European Unions' Horizon 2020 Research and Innovation programme under the Grand Agreement Number 869984.

OPEN_NEXT – Transforming collaborative product creation

Consortium:

#	Participant Legal Name	Short Name	Country
1	TECHNISCHE UNIVERSITAT BERLIN	TUB	DE
2	INSTITUT POLYTECHNIQUE DE GRENOBLE	GINP	FR
3	ALEXANDER VON HUMBOLDT-INSTITUT FURINTERNET UND GESELLSCHAFT GGMBH	HIIG	DE
4	UNIVERSITY OF BATH	UBA	UK
5	ZENTRUM FUR SOZIALE INNOVATION GMBH	ZSI	AT
6	FRAUNHOFER GESELLSCHAFT ZUR FOERDERUNG DER ANGEWANDTEN FORSCHUNG E.V.	FHG	DE
7	DANSK DESIGN CENTER APS	DDC	DK
8	WIKIMEDIA DEUTSCHLAND - GESELLSCHAFT ZUR FÖRDERUNG FREIEN WISSENS EV	WMDE	DE
9	WIKIFACTORY EUROPE SL	WIF	ES
10	STICHTING WAAG SOCIETY	WAAG	NL
11	MAKER	MAK	DK
12	AGILE HEAP EV	FLB	DE
13	SONO MOTORS GMBH	SOM	DE
14	OPNTEC GMBH	OPT	DE
15	STYKKA APS	STY	DK
16	TILL WOLFER	XYZC	DE
17	FICTION FACTORY	FIF	NL
18	M2M4ALL	SOD	NL
19	INNOC OSTERREICHISCHE GESELLSCHAFT FUR INNOVATIVE COMPUTERWISSENSCHAFTEN	HAL	AT

Duration: 09/2019-08/2022
Grant: H2020-869984
Contact (coordinator): Prof Dr-Ing Roland JOCHEM
Address: Technische Universität Berlin, Sekretariat PTZ 3, Pascalstr. 8-9, 10587 Berlin
E-mail: roland.jochem@tu-berlin.de

D1.2 - Quality Framework

Review and approval status:

	Name and Surname	**Role in the Project**	**Partner**
Author(s)	Romain Jego	Researcher	GINP
	Amer Ezoji	Researcher	GINP
	Romain Pinquié	Researcher	GINP
	Jean-François Boujut	WP2 coordinator	GINP
Reviewed by	Cansu Tanrikulu	Researcher	IPK
	Stephan Bohn	Researcher	HIIG
Approved by			

History of changes:

Version	**Date**	**Description of changes**	**By**
0.1	29.07.2022	Initial draft	Romain JEGO
1.2	16.08.2022	Intermediate draft	Romain JEGO
2.0	08.09.2022	Final draft	Romain JEGO
3.0	26.09.2022	Final version	Romain JEGO
4.0	30.09.2022	Final version: reviewed	Romain JEGO

Details:

Dissemination level	Open Access
Due date	
Issue date	
Contract No.	869984
Responsible Partner	GINP
File name	D2.6_ *guidelines for documentation of open-source hardware design reuse*

Executive summary:

This document reports the achievement of the work carried out in the Task 2.3 of work package 2 focusing on models and standards for documentation and knowledge capture enabling reuse of OSH design. Documenting open-source hardware design faces many challenges, yet the documentation phase carries substantial weight in project success. The lack of standardized open-source hardware documentation guidelines and the poor commitment to the documentation task makes the reuse of open designs very problematic.

Based on the Open-source Hardware Association definition, the documentation should ensure that the information is accessible for studying, modifying, or replicating a piece of software or hardware. The aim of the project was to create a minimal documentation enabling design reuse. This report is to present a set of standardized guidelines and templates that have been developed and tested during the OPENNEXT project.

From the analysis of industry standard design processes and existing well-documented projects, we produced the eight different documentation sections covering the whole product life-cycle: Introduction, Specification, Design, Manufacturing, Assembly, Use, Maintenance and Disposal. Structured by these sections, the content of the proposed guidelines incorporates pedagogical and didactic elements to facilitate the construction of the documentation. To be compliant with the designers' practices and be more flexible, we used the concept of development activities. These development activities are related to various product life-cycle phases depending on the design objectives, maturity of the project, etc.. In order to match the different objectives and needs for documentation, we built a modular approach based on the identification of the development activities along the product life-cycle.

The template built around the eight sections of documentations is the result of the synthesis of numerous design practices that we organized into a minimum viable documentation. Two different implementations of the guideline have been published:

- The GitHub guidelines, is complete structure of documentation, in the form of a GitHub template that can be duplicated and populated: https://github.com/OPEN-NEXT/WP2.3-Guideline-and-template-for-documentation-of-OSH-design-reuse
- The flexible implementation enabling a customizable documentation depending on the aim and scope of the designers is published on the community-based Wikifactory platform: https://wikifactory.com/+opennext/projects

The documentation solutions were field-tested during the demonstration phase, through a follow-up process with SMEs and a methodical comparison with existing web-based documentations. This validation of our guidelines is based on the qualitative analysis of the documentation of 3 projects enrolled in the demonstration phase and 6 well-documented web-based open-source development projects.

The workshops and monthly meetings with SMEs organised during the demonstration phase allowed us to capture the perception and understanding of practices for OSH documentation. This demonstration phase included 13 monthly meetings for 6 SMEs between March and May 2022.

The main lesson learnt here is that blockage during the documentation process could be imputed to a lack of strategic questioning regarding openness, especially concerning the business models. The strategic questioning should be addressed before starting the documentation process. The requirements concerning the documentation process in a specific project is derived from this strategic questioning. Eventually, this research led to consider that the process for documenting open-source hardware projects is part of the product design process itself.

The qualitative analysis of well documented projects enabled us to demonstrate that our guidelines sections have been addressed at least once and that our documentation structure was complete. The analysis of projects' criteria of dynamism indicates that the more complete the project is from a documentation point of view, the more dynamic the community tends to be. It emphasizes on the importance of quality of documentation in OSH projects.

These results contribute to promote good documentation practices among SMEs and Makers, further investigation regarding the articulation of the documentation with the different business models and other factors of documentation success need to be carried out. OSH projects should integrate a strategic reflexion on the openness and the level of documentation. This is a condition for the adoption of OSH by SMEs, and necessary to develop industrially viable products that reuse open-source designs.

Table of contents

List of abbreviations and terms

OSH ..... Open-source Hardware
OSHWA ..... Open-source Hardware Association
FBS ..... Function-Behaviour-Structure
OSD ..... Open-source Development
OSP ..... Open-source Process

List of figures

List of tables

1 Introduction

1.1 Motivation

Collaborative product creation entails combined work of different people. Open-source Hardware (OSH) promotes opening the design process one step further, inviting virtually anyone in the world to take part and work on the project. This possibly implies different locations, different timelines, different expertise, different motivations. Thus, the success of collaborative open-source work is correlated to the ability of people to share, receive, learn, add, detailed technical information of the product and its design. The set of information shared on digital support is what we call "documentation". Therefore, a good documentation is a key in the success of open-source projects.

The Open-source movement promotes openness and accessibility essential values. According to the OSHWA definition, documentation is introduced as key for the achievement of the fundamental freedoms of open-source hardware.

"The hardware must be released with documentation including design files, and must allow modification and distribution of the design files. Where documentation is not furnished with the physical product, there must be a well-publicised means of obtaining this documentation for no more than a reasonable reproduction cost, preferably downloading via the Internet without charge. The documentation must include design files in the preferred format for making changes, for example the native file format of a CAD program. Deliberately obfuscated design files are not allowed. Intermediate forms analogous to compiled computer code — such as printer-ready copper artwork from a CAD program — are not allowed as substitutes. The licence may require that the design files are provided in fully-documented, open format(s)."

The freedom to study entails the accessibility of technical information such as the native source files, the requirements, the functional models, etc.[1]

A well-constructed documentation ensures that the information is accessible for studying, modifying, or replicating a piece of software or hardware.

This work is motivated by the very diverse and contrasted documentation practices observed in web-based open-source hardware projects. Indeed, the lack of standardized open-source hardware documentation guidelines, and the poor commitment to the documentation task, makes the reuse of design solutions very problematic. Besides, the motivation for documenting is very connected to the aim and goal of project owners. Depending on the project or business objectives, the documentation needs and contents might be different.

[1] https://www.oshwa.org/definition/

Following the work synthesized in the deliverable 2.1, we hereby report the construction of guidelines and templates and their implementation. This results in a minimum viable design documentation that ambitions to optimize efforts and quality, in accordance with the user's goal in terms of openness.

This deliverable reports the achievement of the work carried out in the Task 2.3 of the work package 2. It is organized into 4 main chapters. Chapter 1 presents the motivation and the methodological approach of the project, chapter 2 details the structure of the guidelines and its foundations inspired by industrial design practices. Chapter 3 describes the implementation of the guidelines in two different community-based open design platforms. Finally, chapter 4 concentrates on the validation made with the OPENNEXT partners during the demonstration phase and the comparison of our guidelines with existing well-documented projects.

1.2 Objective

The main objective of task 2.3 is to deliver a set of guidelines for documenting the design of OSH. The guidelines include a documentation template, standing as the minimum viable documentation for open-source hardware design, which facilitates the completion of documentation. The objective is to support open-source practices and practitioners in their task of capturing and sharing technical information on the design and all the related information (design rationales, technological solutions, manufacturing and assembly instructions, etc.).

The guidelines developed is accessible via the GitHub template. In addition, to facilitate the use of guidelines for documentation, an implementation on Wikifactory is proposed to ease and customize the use of documentation sections according to the needs OSH designers.

Another objective is to field test the guidelines during follow up with SMEs. The goal is both to promote the use of the documentation and to refined our understanding of the challenges faced.A last objective is to validate the completeness of the guideline and question the correlation between the quality of documentation and its repercussion on community dynamic.

1.3 Method

We report here the general method followed to ensure the utility and usability of the guidelines and their implementation.

Step 1 – Set the base of the guidelines

In this context of open-source collaboration, production and retrieval of information are two facets of the documentation use and dynamics. With this in mind, we worked out guidelines that comply with the open-source freedoms define by OSHA (see section 1.1).

The concept of **5 personas** – namely, "Participant in OSD projects", "Inventive creator", "Entrepreneur and OSP project founder", "Community manager & Visionary", "Maker & Geek" - conclusions of "OPEN - Methods and tools for community-based product development", was also used as a main pillar for our work. The persona characterizes a

significant proportion of people in the real-world taking part in open-source project. These ideal types represent the actors who are potentially involved in the creation, modification and use of the documentation.

Step 2 – Preliminary analysis – refine expectations

One core motivation for documenting OSH products is to facilitate the reuse - with or without modification - of existing designs. This, a preliminary analysis aimed at better understand the design reuse problem by identifying design reuse approaches, makers expectations, and design reuse goals (figure 1):

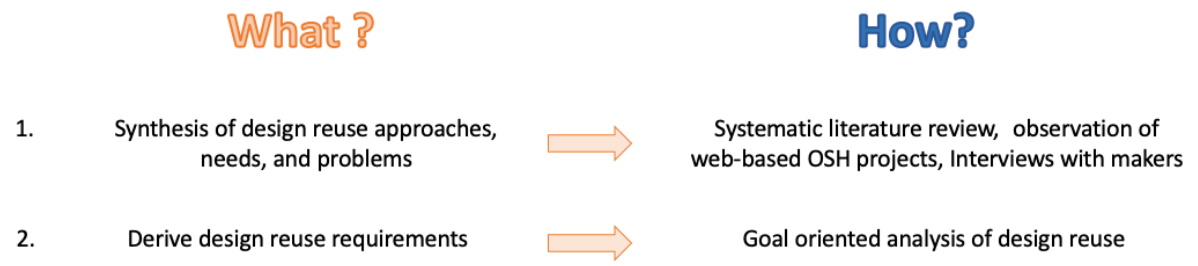


Figure 1: Illustration of the method used to elicit the requirements for documentation

The design reuse approaches, needs and problems were identified through a systematic literature reviews (Ezoji, A., Boujut, J.F. and Pinquié, R., 2021), the study of industry best practices, the analysis of web-based OSH projects, and the interview of OSH makers. From these analyses, we elicited needs for documentation guidelines which are compliant with our conclusions regarding both industrial and maker current practices.

Steps 1 and 2 have been reported in deliverable 2.1: “First release of models and standards for design reuse”.

Step 3 – Creation of documentation guidelines and templates for the design of OSH

Based on the preliminary analysis, we set here the documentation guidelines principles. These principles have been inspired by the knowledge we have from the industrial development processes and follows a phase-based new product development approach. Furthermore, we have considered the activities and goals of the projects owners to come up with an activity-based approach consisting of documentation sections that can be combined to come up with a documentation that suits the needs and maturity of the project. Documentation is therefore considered as a process of continuous improvement that follows the life cycle of the product.

Step 4 – Implementation on GitHub and Wikifactory

In parallel to the creation of the guideline’s categories, we implemented them in the GitHub platform. This implementation allowed to test the feasibility of our guidelines schema and check its validity on existing projects.

After the creation of the GitHub guidelines and template, we collaborated with the staff of Wikifactory who implemented the guidelines on the Wikifactory platform. We implemented the activity-based approach and elicited the documentation sections and the mapping with design activities. The goal was to facilitate the use of the documentation templates and reduce the documentation cost.

Step 5– Tests and validation during the demonstration phase

We have proposed a documentation process monitoring and support during the demonstration phase. This has been the opportunity to support the SMEs in their documentation process and help them to clarify their needs and goals. During this phase we have tested the completeness of our documentation guidelines.

Step 6 – Analyse of well documented project and comparison with our guidelines

In this step, we wanted to verify that our guidelines and propose some good practices for makers to better document there projects according to their objectives. This resulted in the analysis of several existing projects and a comparison with the framework of our guidelines.

2 Creation of documentation standards, guidelines and templates

2.1 Objective

The objective is to create and make accessible minimum viable design documentation guidelines for open-source hardware project. The first activity aimed at reviewing and selecting the different parts constituting such documentation. This review relies on our previous studies (see section 1.3 and first deliverable). The proposed documentation procedure enables OSH makers to create their documentation based on solid comprehensive work.

2.2 Method

As stated in deliverable D2.1, our aim was to develop the guidelines that would allow the three-fundamental freedom of OSH, (i.e. study, make, modify) to be possible. We expected to derived those freedom into goals that would guide the action of the creators of the documentation. This approach was kept as a frame for our work.

A second source of inspiration was the extensive knowledge describing the industrial New Product Development (NPD) processes. Seminal works such as Stage-gate models (Cooper and Kleinschmidt, 1988), V shaped models[2], or more recently the design thinking model[3] led us to propose the basic structure presented in figure 2.

[2] https://en.wikipedia.org/wiki/V-Model
[3] https://en.wikipedia.org/wiki/Design_thinking

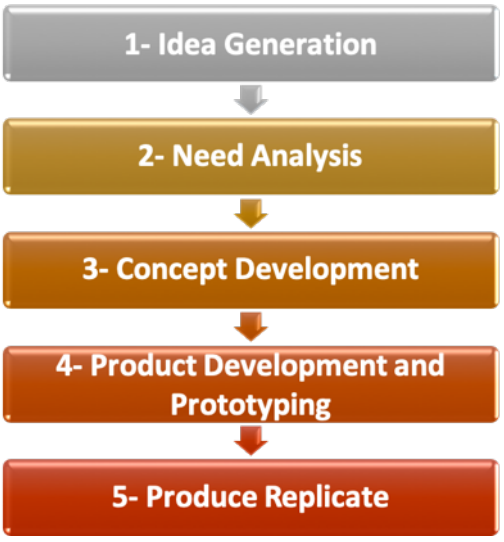


Figure 2: Prototypical schema of a design process

Knowledge-based approaches recognize design as a collaborative and knowledge intensive activity. During design, a product is a set of models, and depending on the perspective of a designer, these models may be very diverse. They can describe the product in its context of use with very nice realistic renderings, or use very abstract forms like equations and charts. A third source of inspiration comes from the work of Gero and al. who consider that products are represented by functional, behavioural or structural models (Dorst and P. E. Vermaas, 2005).

Today, this is an agreed-upon product design ontology named Function-Behaviour-Structure (FBS) where:

- Function (F) stands for “what the object is for”
- Behaviour (B) stands for “what the object does”
- Structure (S) stands for “what the object consists of”

Function is about the relation to the use, usage and context. Behaviour is about how the product evolves and responds to external stimuli, which is represented by diverse kinematic and dynamic simulation models. Structure is about how the product is embodied into a physical element, which includes 3D models, layout etc. We selected FBS as the most suitable design model to structure and capture the core of the product model description in the template.

In addition to the scientific literature approach, we searched for well documented open-source projects (see below). These existing projects have been analysed with the lens of the previously presented work. We also used these projects as examples in the documentation guidelines.

- BCN3D Moveo Project - section "Motivation and goal in the introduction"
- NASA JPL Open-source Rover Project - section "Skills Necessary, Software description, Manufacturing, Assembly"

- Open-Source Ventilator Project - section "Functional Diagram"
- Renesas Electronics Ventilator Project - section "Functional Diagram"
- GlasVent Project - section "Diagram of Architectural Structure"
- GMPS Open-Source Ventilator Project - section "Hardware overview"
- MakAir Covid-19 Ventilator Project - section "Architecture"
- Echopen - section “introduction” and “use”

2.3 Development

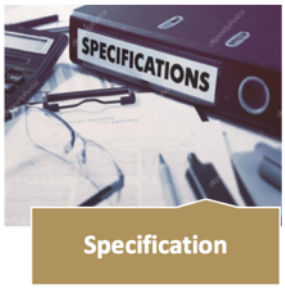
2.3.1 Creation of the different sections of guidelines for documentation

Based on a methodological study of industry standard design process and inspiration from well-documented projects, the analysis allowed us to identify the different sections to cover the whole product life cycle.

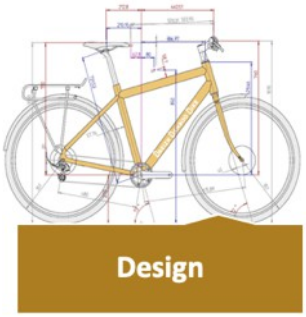
Our approach of the documentation is product-centric as the vast majority of the documentations encountered throughout our study. The framework of the guidelines is therefore following (or copying) the development process of the product, from early design phases throughout the life of the product until the disposal phase.

The guidelines consist of eight independent chapters, each one documenting a specific development phase. A brief overview of these chapters is presented below:

The introduction to the document aims to provide more details about the project goal, and the context. This introduction should allow the reader to first understand the motivation of the project and also who is involved or has been involved in its creation. As the documentation aims at fostering the reuse of the technical contents, it is important to understand the context.

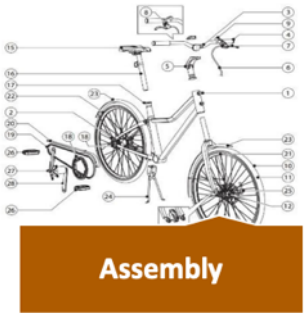


The specification chapter consists in describing the product with a "black box" external view to capture the intermediate technical objects (stakeholders, external interfaces, services, and constraints) that serve to, in fine, elucidate the product requirements.



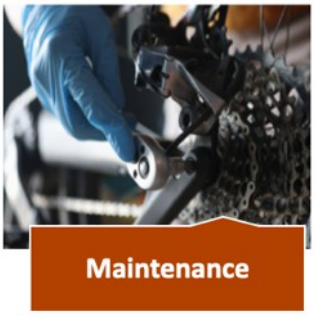
The design chapter refers to the development of one or several solutions to the design problem defined by the specification. In this section, we find mainly models of the product under various forms. The product models provide representations of the concepts being developed. Product models serve to answer questions that interest multiple views on the product, we choose to use the Function-Behaviour-Structure (FBS) views.

The manufacturing chapter aims to guide the makers through the process of making the product. Manufacturing instructions mean full description and instructions concerning raw material, operating conditions, and process to be employed for the manufacture and assembly of the product.

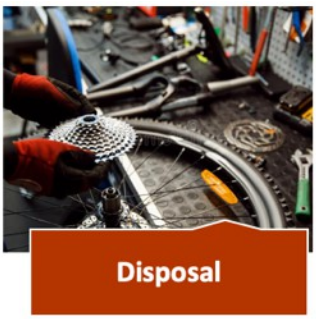


The assembly chapter includes the instructions that guide the makers throughout the process of assembly or disassembly of the components. So, the makers can understand the steps for constructing a finished product from components or partially compiled units.

The use section includes the guide which consists of information that enables the end-users to operate the product, it helps non-technical people pinpoint and solve simple problems without expert assistance. The use section may also include troubleshooting sections.



The maintenance chapter helps the makers to understand the process for repairing the system by replacing components to correct defaults and ensure a satisfactory life time. Modifications can also concern the improvement of the performance or other attributes, or adapt the system to a changing operational environment.



The last phase of a product life cycle, the disposal, documents how makers should discard some or all parts.

A systematic structure of each chapter has been set up. To be as transparent as possible, we have inserted a "what" section presenting the chapter's topic, a "why" structure presenting the goal of the chapter. Then we have a "how" section that describes the guidelines and templates. Finally, for each template, we have examples of documentation of this section. This structure has been systematically implemented with local variations related to the specific needs of the chapters.

2.3.2 Pedagogical dimension of the guidelines

The aim of the documentation structure is to supply a guidance for organizing the information during the documentation process. The associated templates rely pedagogical and didactic elements to facilitate the use and filling of the documentation. The guidelines start with an overview of our work, including an introduction to documentation challenges, exposing general purposes of the template and describing core element, followed by a brief getting started section. The structure of the documentation itself (folders tree) embeds some knowledge related to the product development process of the artefacts. The systematic structure of the content is presented below:

3. Service *(synonyms: external function, capability)*

▼ Click to see the guideline!

- **Definition:** *A service is an effect intended by a stakeholder resulting from the interaction of the product with its environment (i.e. what the product is for).*
- **Comments:**
  - *Services provide users with an exchange value that can be included in an economic system (e.g. airlines buy flight hours).*
  - *Services are intended effects that can be observed from outside the product ("black box" external view), but not from outside an internal component ("white box" internal view).*
  - *Services are defined in a solution neutral-way.*
  - *Services can be stated as follows: The [Product] shall enable [Stakeholder] [Action verb] (e.g. The product shall enable end-user to clean its teeth)*
  - *we often reason in terms of action verbs to communicate expected behaviors, so it would be nice to be able to search designs with action verbs*

▶ Click to see the example!

Template of services

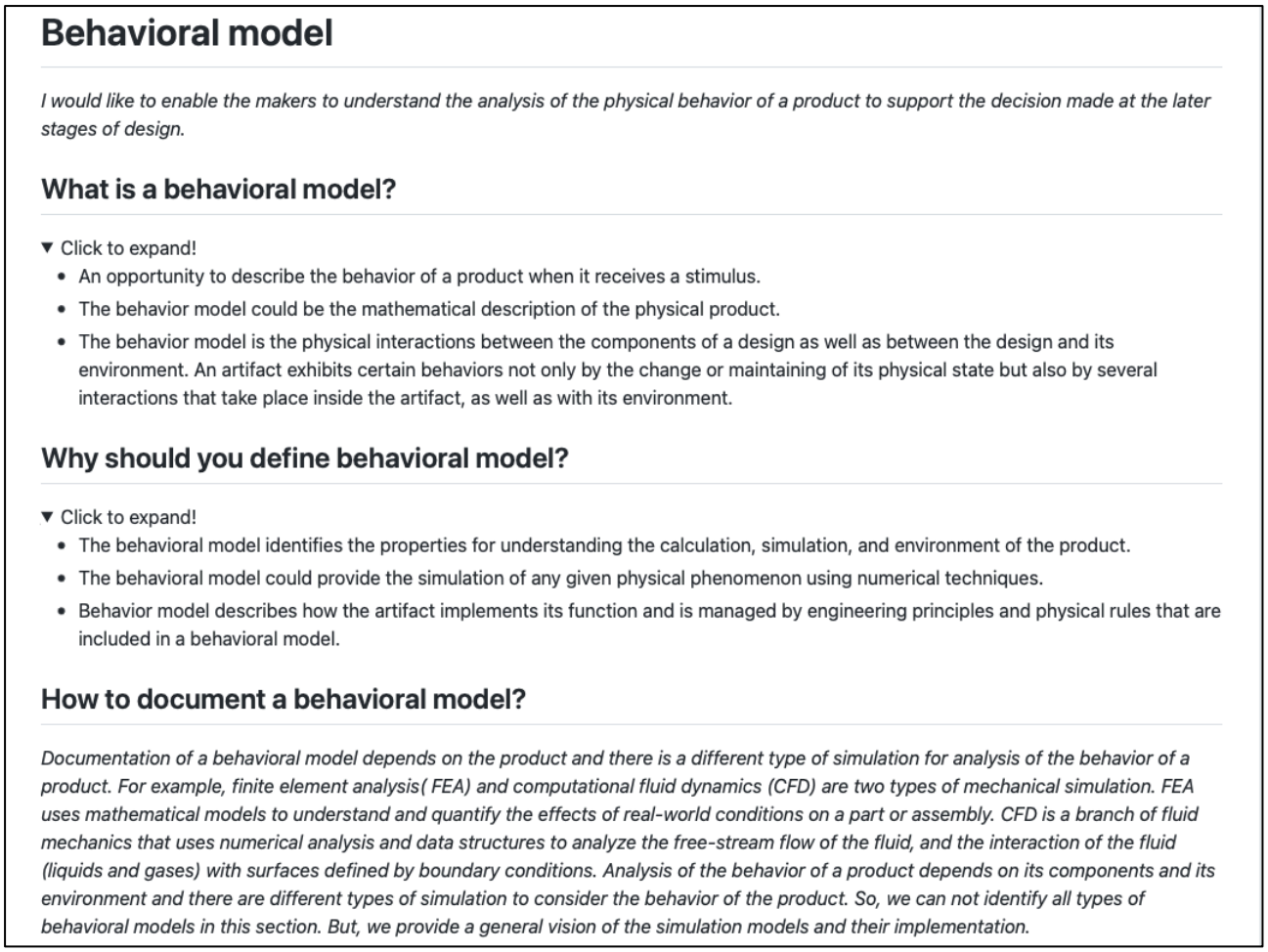
- Services to stakeholder 1
  - Service 1.1

Figure 3: Example of section "service" composed of a definition, comments, example and template

https://github.com/OPEN-NEXT/WP2.3-Guideline-and-template-for-documentation-of-OSH-design-reuse/tree/main/Documentation/2.%20Specification

The contextual elements have a function of providing information that explains why such solution is preferred. This is an important factor that relates to the notion of design rationale. Design rationale entails all elements of information that allow the users of the information to understand the why and how of a decision. To enable design reuse, meaning the reuse of the documented solutions, the presence of elements of rationale is important.

Another dimension is the "how" to document. In fact, most of the makers do not really have the knowledge of all the modelling tools or the level of information relevant to provide. There is a clear need to provide guidance to support the makers in the construction of the documentation. Each part of the template comes along with a text that explains the aim of the chapter, and what to expect in this section (figure 3). Each section offered precise template pre-filled with examples to illustrate what could be expected (figure 4). For each part, the questions "what is this section for?", "Why do I need this in the documentation?" and "How should I document this part?" are covered.



Behavioral model

I would like to enable the makers to understand the analysis of the physical behavior of a product to support the decision made at the later stages of design.

What is a behavioral model?

▼ Click to expand!

- An opportunity to describe the behavior of a product when it receives a stimulus.
- The behavior model could be the mathematical description of the physical product.
- The behavior model is the physical interactions between the components of a design as well as between the design and its environment. An artifact exhibits certain behaviors not only by the change or maintaining of its physical state but also by several interactions that take place inside the artifact, as well as with its environment.

Why should you define behavioral model?

▼ Click to expand!

- The behavioral model identifies the properties for understanding the calculation, simulation, and environment of the product.
- The behavioral model could provide the simulation of any given physical phenomenon using numerical techniques.
- Behavior model describes how the artifact implements its function and is managed by engineering principles and physical rules that are included in a behavioral model.

How to document a behavioral model?

Documentation of a behavioral model depends on the product and there is a different type of simulation for analysis of the behavior of a product. For example, finite element analysis(FEA) and computational fluid dynamics (CFD) are two types of mechanical simulation. FEA uses mathematical models to understand and quantify the effects of real-world conditions on a part or assembly. CFD is a branch of fluid mechanics that uses numerical analysis and data structures to analyze the free-stream flow of the fluid, and the interaction of the fluid (liquids and gases) with surfaces defined by boundary conditions. Analysis of the behavior of a product depends on its components and its environment and there are different types of simulation to consider the behavior of the product. So, we can not identify all types of behavioral models in this section. But, we provide a general vision of the simulation models and their implementation.

Figure 4: Screenshot of thorough explanation regarding the Behavioural model section in the GitHub template

https://github.com/OPEN-NEXT/WP2.3-Guideline-and-template-for-documentation-of-OSH-design-reuse/tree/main/Documentation/3.%20Design/Behavioral%20model

Previous interviews conducted with makers have brought to light practical challenges faced when documenting a project. In order to facilitate the user journey through the documentation process, we developed precise template, pre-filled with examples (see figure 5 and 6).

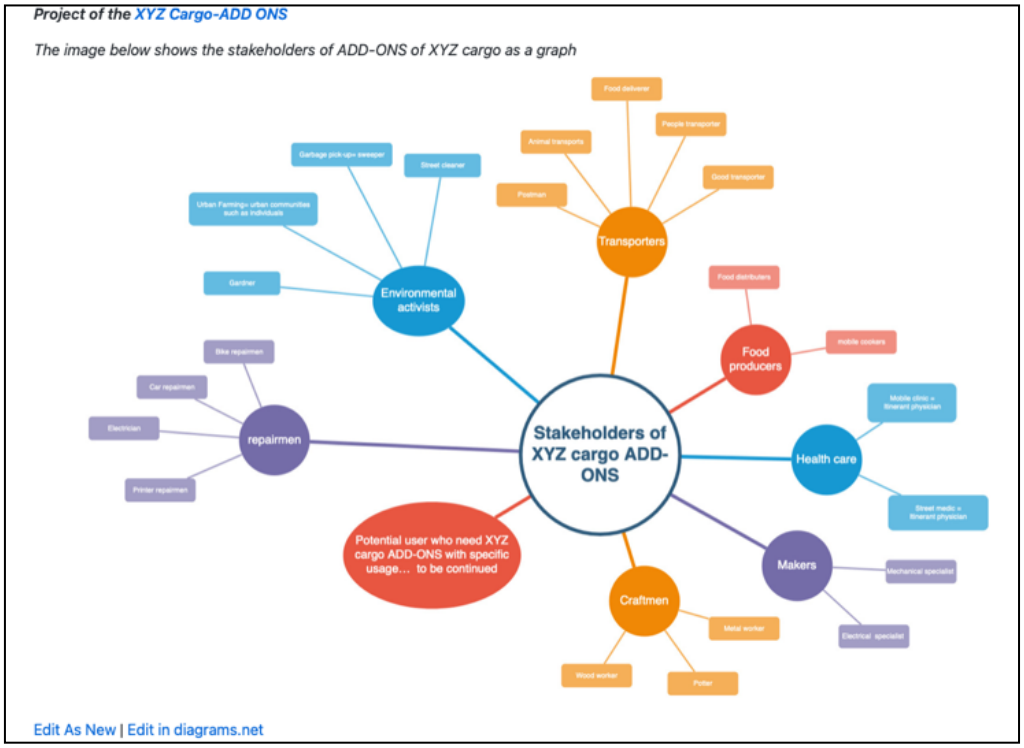


Figure 5: Example of a graph representing a registry of stakeholders that directly interact or indirectly influence the ADD-ONS of the XYZ Cargo. This is part of the specification section in the documentation.

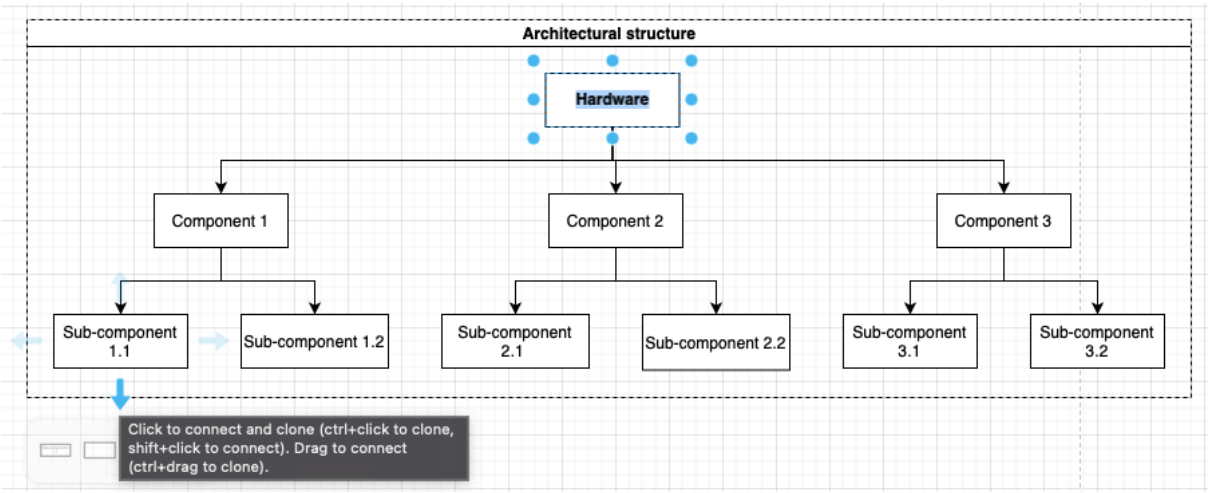


Figure 6: Modification of text, table, tree or graph on App diagram.

https://github.com/OPEN-NEXT/WP2.3-Guideline-and-template-for-documentation-of-OSH-design-reuse/tree/main/Documentation/3.%20Design/Structural%20model/Mechanics

We also integrated various relevant graphs with the possibility to modify them and add any other relevant files. Links to App diagram (https://app.diagrams.net/) in the template provide

the opportunity for makers to fill in the information of their product and create representations of graphs and trees of their product (figure 7).

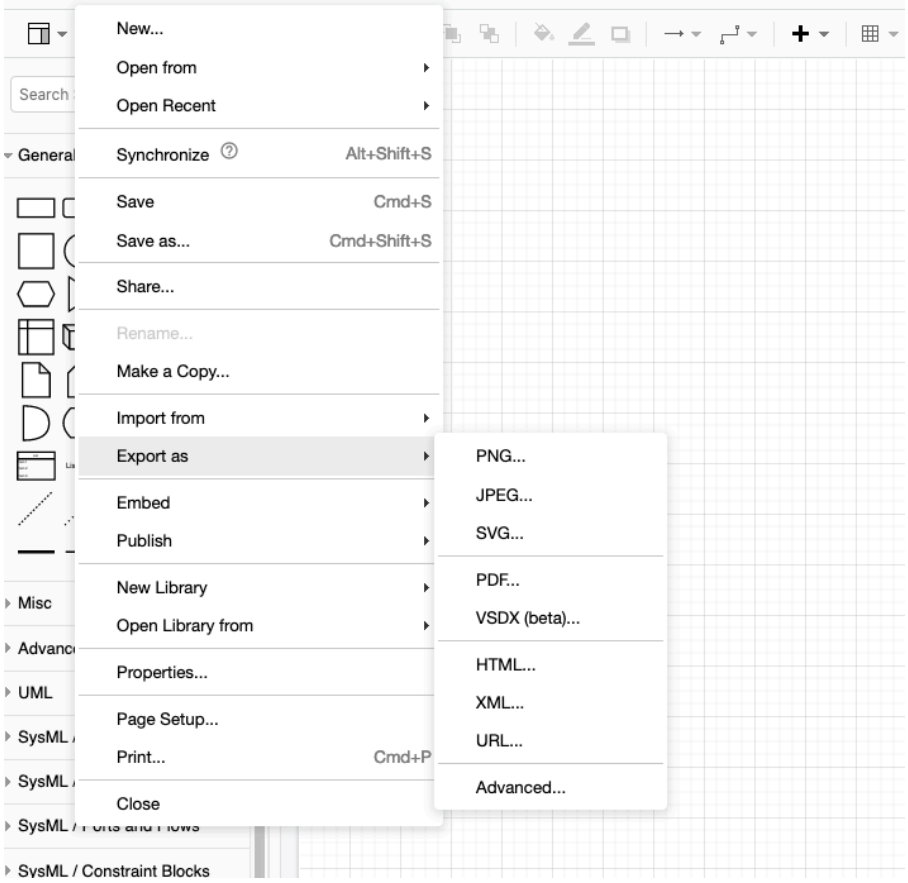


Figure 7: Export the different file formats of App diagram

2.3.3 **Analysis and discussion**

The method followed to create the guidelines sections and their content relies on the review of design reuse approaches and the analysis of existing good documentation practices. The review of design reuse approaches enabled us to capture the most important factors influencing design reuse (see deliverable D2): product model, design rationale, communication tools, software support. The two first factors have been addressed in our approach. product model and design rationale, both endorsed by scientific literature but forsake by real world documentation. Both contribute to design reuse and can be identified through the content of the guidelines as previously presented.

For example, Figure 8 and 9 show two functional trees that represent the systematic decomposition or the functions of a technical system (see FAIREMBO project[4]). The analysis

[4] https://github.com/TomGosnik/FairEmbo-Project

of these trees enables the reader to understand the reasons why the system is made of two sub-systems and the sub functions they address.

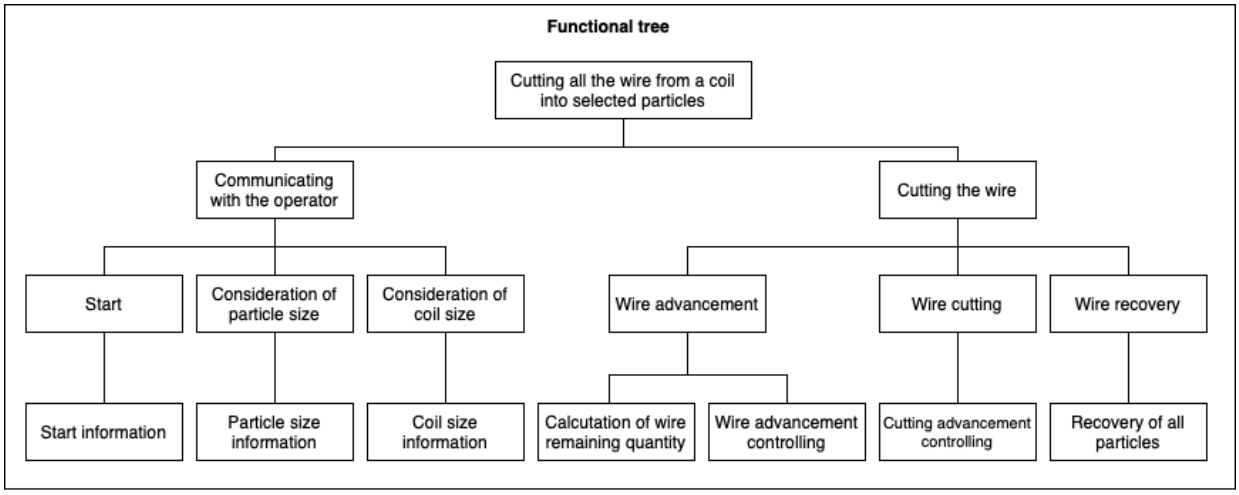


Figure 8: Example of functional tree diagram from FairEmbo project's documentation

https://github.com/TomGosnik/FairEmbo-Project/blob/main/Documentation/3.%20Design/Functional%20model/Functional%20model.md

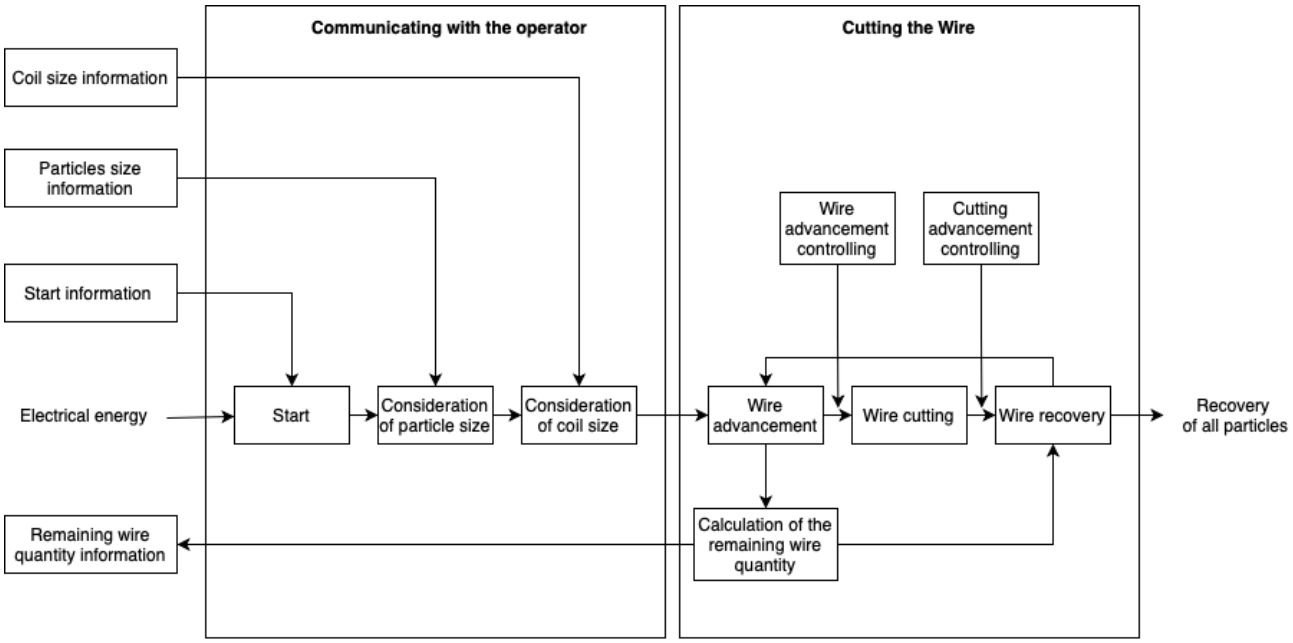


Figure 9: Functional graph of FairEmbo solution

https://github.com/TomGosnik/FairEmbo-Project/blob/main/Documentation/3.%20Design/Functional%20model/Functional%20model.md

The analysis of the template enables us to evaluate design rationale presence in the guidelines. The template built in and explanatory texts pushes to make explicit design assumptions, especially the rationality of design choices. Thus, it brings into existence a form of design rationale that can be retrieved through the analysis of the documentation. Additionally, the representation figure 8 and 9 are instantiation of the product model from a functional perspective.

2.3.4 Definition of the development activities

We have observed that the documentation sections constituting the guidelines (see 2.3.1) could be more flexible with regards to designers' expectations, that is, designers could select pre-formatted documentation building sections and custom the organization thanks to modularity. At first, we sought to define these generic documentation requirements to meet designers' generic needs. In our work, the "development activities" are specific activities tied with specific product life cycle phase. Thus, development activities specify OSH designers needs for documentation, specifically related to a specific product life-cycle phase.

We considered the product life-cycle spread across eight generics phases, namely:

1- Idea generation
2- Need analysis
3- Concept development
4- Product development and prototyping
5- Produce/ replicate
6- Use
7- Maintenance
8- Dismantle/ recycle

The process of the product life cycle starts with "idea generation" and "need analysis", followed by "concept development" to solve a problem. The "Product development and prototyping" phase consists of developing the selected concept. After the designing phase, the prototyping phase is an early sample, model, or release of proof of concept. Then, the product can be manufactured. The three last phases simplify the procedure of usage, maintenance of product, and the process of removing the product or its parts for ensuring the proper handling of any environmentally sensitive, or valuable materials.

Our method for identifying the development activities is based on previous studies comporting literature reviews, observation of web-based OSH projects, and interviews with makers. Based on this study, we identified and defined the development activities. We illustrate this with an example, the PiKon telescope, an open-source 3D printed, Raspberry Pi-powered Astro-cam.

During an interview, the developer of PiKon said that he gathered a community of science and space enthusiasts from around the world. Moreover, a repository for the PiKon telescope enables the members of the PiKon community to share design data and information. Indeed, this repository can help the makers by *sharing the results and experiences* to find the *needs* leading to new design problems. We considered that *problem discovery*, as a development activity, falling into a wider category of *information search* which is synthesizing information to elicit a question, a problem etc.

We identified the following 12 different development activities; their definitions consist of:

1- ***Information search:*** aims to gain an understanding on the problem that you're trying to solve. So, it is necessary to search for information that help understanding a context, find inspiration, problems to solve, examples, existing designs, illustrations, etc.
2- ***Problem description:*** In this activity you describe the context of your project. The formulation of the problem entails the description of the situation where the users encounters a problem. This formulation can take the form of a text, photos or videos. The important is that the reader understands what is the project about and why this project is developed.
3- ***Idea contribution:*** This activity relates to idea generation. Contributors are welcome to propose new concepts or any new idea to contribute to the project. This can consist at looking for alternative ways to consider the problem and identify innovative solutions. The contribution can be very diverse, it can be on the form of sketch, photos, text, etc...
4- ***Benchmark:*** For this activity, the documentation should allow to carry out classical benchmarking activities. Users could be able to seek for other similar products, focusing on diverse aspects of the documentation.
5- ***Specification:*** This activity consists of capturing the functional and non-functional requirements of the product. The specification process requires to capture intermediate information including the stakeholders, the external interfaces between the product and the stakeholders, the use cases, the needs, etc.
6- ***Technical contribution*****:** Also known as detailed design, the technical contribution contributes to describe the essentials of technical specifications, concepts, solutions and detailed implementation of the product. This activity follows the functional specification. Here all the technical developments should be documented, this includes models, simulations, CAD models, etc...
7- ***Prototyping and testing the product:*** This activity should be documented with manufacturing information, Bill of Material, assembly instructions, test results, and all useful information that allows the improvement of the solution. The purpose of prototyping and testing the product is to test the functionality of the product and make it actually work in its operational context while ensuring the safety and quality of use.
8- ***Customization of design:*** The activity relates to the appropriation and the modification of an existing design. Here you can document a branch of the initial design with your modifications.
9- ***Replicate/ Produce:*** At the core of the open-source hardware definition lies the freedom for anyone to replicate the hardware based on the design. Replicability, i.e. the ability of you to build a functioning version of the hardware in their location is of paramount importance to OSH. In this activity you should provide documents that easily allow replication of the product, including assembly, manufacturing...
10- ***Using the product:*** In this activity one should document the user guide that enables end-users to operate the product properly, it helps non-technical people pinpoint and solve simple problems without expert assistance during of use phase.
11- ***Service / Maintenance:*** Maintenance instructions should provide the necessary information to maintain the system effectively and to perform the required maintenance operations so that the system works properly in the long run. This includes the recommendations on the frequency of the maintenance and the risks of failure. The

purpose is to enable regular and inexpensive service and maintenance by providing documents for replicating components, identifying troubleshooting...

12- ***Dismantle for recycling or disposing:*** The purpose of the disposal process is to remove a system element from the operating environment with the intent of permanent termination of its use. This process can deal with any hazardous or toxic materials or waste products following the applicable guidance, policy, regulation, and statutes. This activity should provide information for properly disassemble the product and recycle the components.

2.3.5 Identification of needed documentation sections according to development activities, users, and fundamental goals of OSH

As mentioned in the previous section, the development activities aim at specifying sub-sets of the requirements for documentation according to the contributors' objectives. To achieve this aim, it is necessary to identify the corresponding documentation sections, the user's profiles or motivations and goal in the project. We have matched the different categories to verify that our documentation sections cover all the freedom for OSH, which section covers what freedom, which motivation and actor would preferably be concerned by which section and finally which documentation section matches which activity.

Matching documentation sections and development activities– In table 1 below, we identify the needed documentation sections for each development activity.

		Documentation sections							
		Introduction	Specification	Design	Manufacturing	Assembly	Use	Maintenance	Disposal
Development activities	1. Information search	x	x	x	x	x			
	2. Problem discovery	x	x						
	3. Idea contribution	x	x	x					
	4. Benchmark	x	x	x	x	x			
	5. Specification and requirement collection	x	x	x					
	6. Technical contribution	x	x	x	x	x			
	7. Prototyping and testing the product	x	x	x	x	x			
	8. Customization of design	x	x	x	x	x			
	9. Replicate/ Produce	x			x	x			
	10. Using the product	x					x		
	11. Service / Maintenance			x	x	x		x	
	12. Dismantle for recycling or disposing				x	x			x

Table 1: Identification of required documentation sections of guidelines for each development activity

Matching development activities and freedom for OSH – In table 2, we match the freedom with the development activities defined in section 2.3.4.

		Freedom of OSH			
		Study	Modify	Make	Distribute
Development activities	1. Information search	x	x	x	
	2. Problem discovery	x			
	3. Idea contribution	x			
	4. Benchmark	x	x	x	x
	5. Specification and requirement collection	x	x		
	6. Technical contribution	x	x	x	
	7. Prototyping and testing the product	x	x	x	
	8. Customization of design	x	x	x	x
	9. Replicate/ Produce	x		x	x
	10. Using the product	x			x
	11. Service / Maintenance	x			x
	12. Dismantle for recycling or disposing	x			x

Table 2: Fundamental freedom of OSH covered by each development activity

Matching development activities, documentation sections and associated freedom –To obtain the needed documentation section of guidelines for each development activity, we merged the two previous tables in a single one. To conclude, table 3 sums up the stakeholder's profiles, the freedom, and the corresponding documentation sections of the guidelines.

These results can facilitate the choice of required documentation sections of the guidelines and therefore reduce the effort to put in documenting while ensuring a proper coverage of the documentation. This also allows a progressive approach to the documentation that follows the development process.

Development activities that require creating information	Documentation sections for creating information	Freedom of OSH			
Problem description	D1.Introduction D2.Specification	Study			
Idea contribution	D1.Introduction D2.Specification D3.Design	Study			
Functional specification and requirement collection	D1.Introduction D2.Specification D3.Design	Study	Modify		
Technical contribution	D1.Introduction D2.Specification D4.Manufacturing D5.Assembly	Study	Modify	Make	
Prototyping and testing the product	D1.Introduction D2.Specification D3.Design D4.Manufacturing D5.Assembly	Study	Modify	Make	
Customization of design	D1.Introduction D2.Specification D3.Design D4.Manufacturing D5.Assembly	Study	Modify	Make	Distribute
Replicate/ Produce	D1.Introduction D4.Manufacturing D5.Assembly	Study		Make	Distribute
Using the product	D1.Introduction D6.Use	Study			Distribute
Service / Maintenance	D3.Design D4.Manufacturing D5.Assembly D7.Maintenance	Study			Distribute
Dismantle for recycling or disposing	D4.Manufacturing D5.Assembly D8.Disposal	Study			Distribute

Table 3: Fundamental goals of OSH and Documentation sections for each development activities

While built as a whole to be as complete as possible, the documentation guidelines are in theory modular if required. The following section present two different implementations of the guidelines on two online and community-based platforms.

3 Implementation

We documented in section 2.3.1 the creation of the documentation template containing eight sections. This template is the result of the synthesis of numerous design practices that we organized into a minimum viable documentation. Although the selection of sections is possible depending on the projects' objectives and allows the adaptation of the documentation to the project's requirements, not all online OSH platforms provide the capabilities to tailor the documentation structure and content. Open-source hardware platforms have particular strength and are suitable for specific goals and this chapter describes two different implementations, on GitHub and on Wikifactory.

3.1 Creation of GitHub template

To make the content of the guidelines available for all users, we created a template on GitHub that consisted of a complete set of the documentation sections. This GitHub template is hosted as a repository of the OPENNEXT Project on GitHub[5]. The choice of GitHub as a platform was motivated by several facts. GitHub is a distributed version controller and has strong source code management capabilities. Reasons why there was no discussion about whether or not to put the documentation template on GitHub is because the platform is essential in the Open-Source landscape. GitHub is the first platform popularity-wise for makers, first place to find documentation for collaborative project, code or not. Moreover, this platform facilitates collaborative work notably the project management side. For instance, the issue tracking system allows to distribute very openly tasks, including documentation ones, and easily track the tasks status. GitHub is reliable and the latest version of the code, product design and documentation is always available for the community. To reuse, modify, share, work in collaboration with other on the same files of a project, the "clone" and "fork" function of GitHub become especially convenient. These functions allow working on someone else project. Push and Pull changes depending on method used and repositories ownership. This can be applied to the documentation project itself as well.

Another characteristic of GitHub is the Markdown interpreted language. Markdown is a lightweight and easy-to-use syntax for styling all forms of writing on the GitHub platform. Markdown is a very flexible way to style text on the web. This is an agnostic standard that is independent from any platform. The user controls the display of the document; formatting words as bold or italic, adding images, and creating lists are just a few of the things with Markdown. Mostly, Markdown is just regular text with a few non-alphabetic characters thrown in, like # or *. These characteristics facilitate its use by novice makers in a collaborative and version-controlled environment.

Considering these GitHub strengths, we created on GitHub a repository containing the eight chapters (Figure 10) pre-filled with examples.

[5] https://github.com/OPEN-NEXT/WP2.3-Guideline-and-template-for-documentation-of-OSH-design-reuse

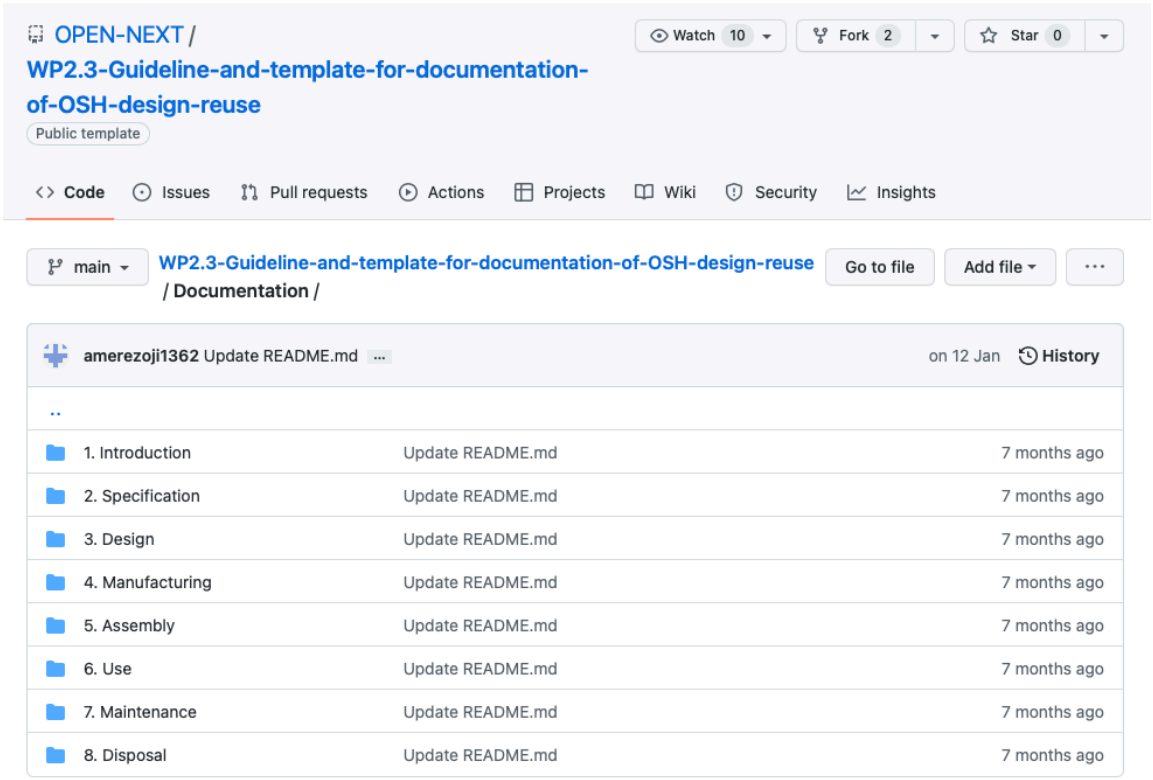


Figure 10: Screenshot of template structure on GitHub

https://github.com/OPEN-NEXT/WP2.3-Guideline-and-template-for-documentation-of-OSH-design-reuse

This repository is edited as a public template and therefore can easily be duplicated and reused using the command <use this template> (Figure 11)

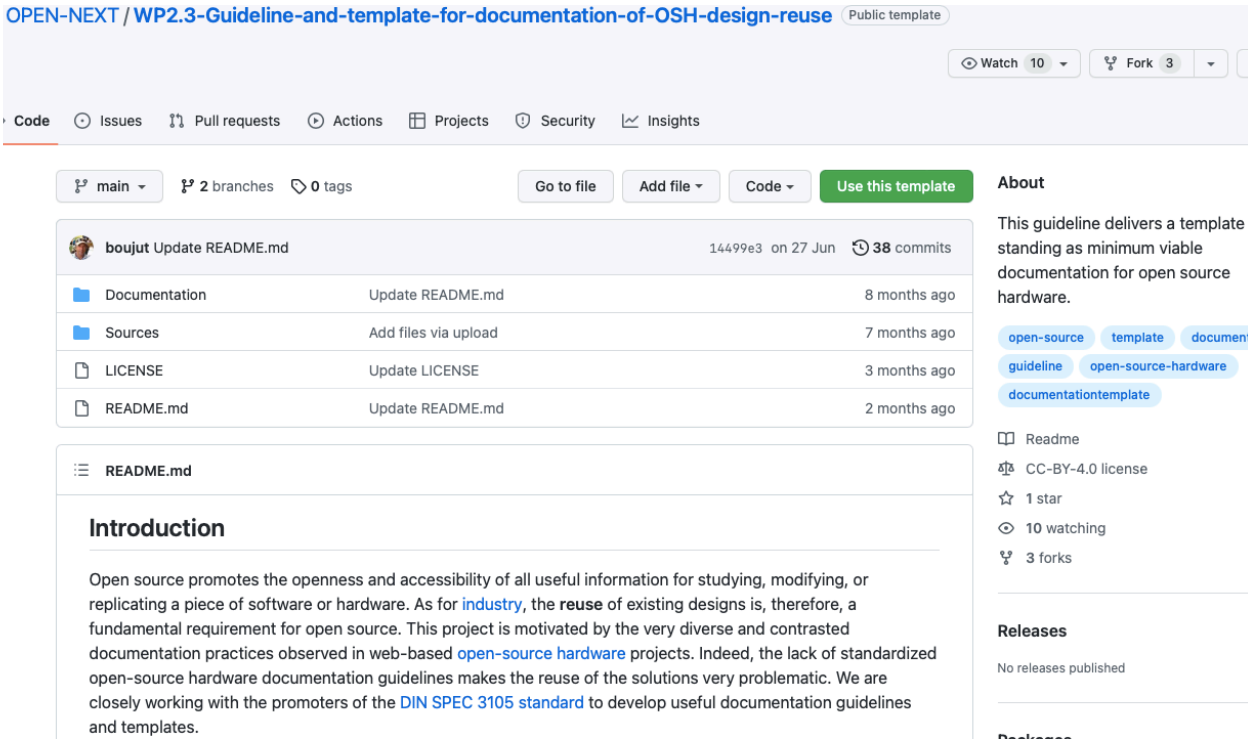


Figure 11: Front page of the documentation template on GitHub

https://github.com/OPEN-NEXT/WP2.3-Guideline-and-template-for-documentation-of-OSH-design-reuse

3.2 Implementation of the guidelines on Wikifactory platform

The GitHub guidelines configuration explores a rather thorough documentation solution, and might not be suitable as is for OSH designers because it does not guide the users through the process of selecting proper documentation sections. Therefore, the objective of this section is to methodically relate the work done implementing documentation process constraint to the designer expectation by implementing the activity model presented below. Also, to make the documentation process accessible, explicit and relevant.

Based on our work to identify needed documentation sections according to development activities, user profiles, and fundamental goals of OSH, described in previous sections 2.3.2 and 2.3.3, the focus was set around the OSH designer's requirements. Often implicit and ill-defined, we needed to bring them out, in order to translate it in documentation section needs. We then use designers' requirements as an entry point of documentation section selection.

Figure 12: Synthetic scheme illustrating process leading to adapted documentation

This section describes the implementation of the activity-based approach that allows the adaptation of the documentation process to match real world requirements and therefore boost the project documentation. In a nutshell, it makes the documentation guidelines flexible but in a relevant manner with a methodological guidance.

Acknowledging this customization need, the OPEN!NEXT project partner Wikifactory implemented our documentation guidelines on their platform. The Wikifactory platform hosts and displays open-source hardware projects, encouraging collaboration, connecting designers, inventors, makers and people, aiming at identifying and connecting competencies, making easier to find the right set of skills-contributor. It also provides CAD and manufacturing support, templates and guides for good development practices. Documentation is one of these services.

The Wikifactory implementation enabled us to harvest the customizable capabilities of the platform and therefore field test the selection of section for documenting OSH project.

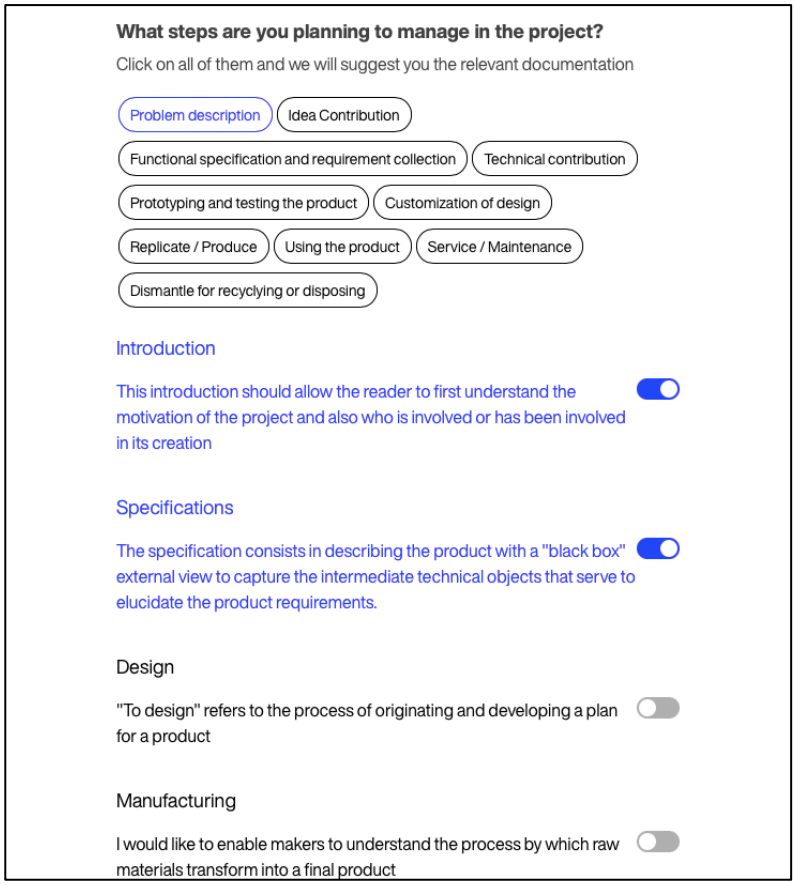


Figure 13: Screenshot of documentation section selection, depending on user goal, on Wikifactory https://wikifactory.com/+opennext/projects

As seen in the above user interface depiction, the implementation allows customizable documentation sections according to the user needs. The interface facilitates the documentation process and therefore promotes good documentation practices. It should be noted that from different requirements, which are for instance induced by different goals or a different business model, derives different documentation outputs.

This research project led to consider the process of documenting open-source hardware projects as part of product design process itself. Different documentation needs derive from different design and projects goals, and result in different outcomes regarding product and project information. The main message to the maker should eventually be: Documenting is part of the design process and should be considered as a task in its own right.

4 Analysis and validation

4.1 Case study validation and demonstration phase

After creating the template and implemented it on GitHub and Wikifactory to enhance usability, and disseminate the documentation practices to SMEs and makers, this section describes our contribution to the demonstration phase.

To get a better insight into the understanding of the documentation structure and the objectives from the SME's point of view, we carried out two workshops and monthly meetings with seven

SMEs and OSH founders. It enabled us to present our work, create momentum on documentation and provide support during documentation phase.

During the two workshops, we presented the content as follows:

- Introduction
  - Objective of workshop
  - SMEs presentation
  - A video of presentation of guidelines and templates of documentation of OSH[6]
  - First Exchange
- Why do we need guidelines and templates for documentation of OSH projects?
- Introduction to the documentation tools and process
  - Clarification of the documentation objectives
  - Presentation of the guidelines and templates philosophy and structure
  - Second exchange time
  - Presentation of the documentation tools and approaches (GitHub and Wikifactory)
  - Third exchange time
- Action plan
  - Fourth exchange time

The participants in these workshops were:

- AVATAR mobilite (https://avatarmobilite.com/)
- Manyone (https://manyone.com/)
- Stroem (https://stroem.readthedocs.io/en/latest/)
- Oskars (https://oskars.org/)
- Ourkilo (https://ourkilo.eu/)
- Anavi (https://anavi.technology/#home)
- Spree Berlin (http://jakobkukula.com/design/spree-berlin.html)
- Playtronica (https://medium.com/playtronica/evolution-of-a-music-instrument-orbita8fe4b0c42913

The workshops enabled us to advocate needs for good documentation practices, present the philosophy and structure of guidelines, and tools and approaches for documentation on GitHub and Wikifactory platforms. A video entitled “guidelines and Templates Enabling Reuse of Open-Source Hardware Design » was presented to the audience and commented[3].

[6] https://www.youtube.com/watch?v=tC9kBOpjx14

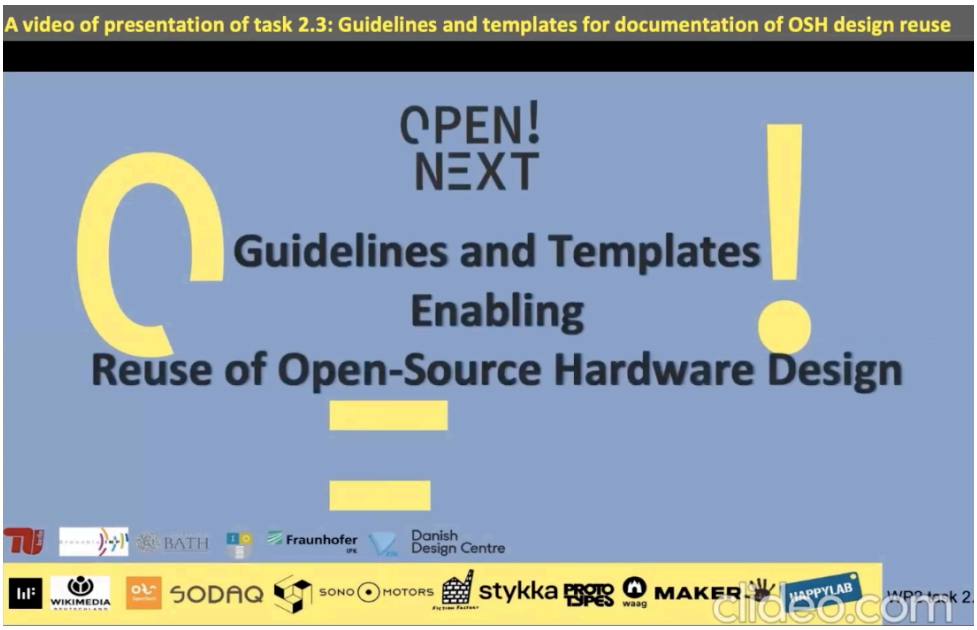


Figure 14: First frame of OSH design video
https://www.youtube.com/watch?v=tC9kBOpjx14

We created this video to easily set an overview regarding reuse of open-source hardware design. This didactic dissemination tool set the context and explains our documentation. The five minutes cover the following sections:

- OSHA statement,
- why reuse design?,
- design reuse approaches in comparison,
- obstacles,
- proposition of our guidelines for design reuse,
- describing the template and platforms.

Moreover, the exchange times during the workshops facilitated the capture of essential answers from the participants, portraying their documentation awareness in open-source project.

The questions discussed with SMEs during the workshop were:

I- What is the aim of your project? What are your expectations?

II- What are the main activities and lifecycle phases (design, manufacturing, user support, distribution...) of the project you are developing at that time (in the demonstrations phase)? What level of openness (open to modification, replication, commercialization, study...) do you want to achieve?

III- What are your needs in terms of tools and support for each project?

IV- How to sketch out action plan for the demonstration phase?

We used Miro Board to collectively answer this set of questions and discuss the answers together. The screen-shot below provides an overview of Miro board used in the workshop. Each participant had a column comporting the questions structuring the reflexion and talks, whereas answers were provided by placing virtual post it notes on the board.

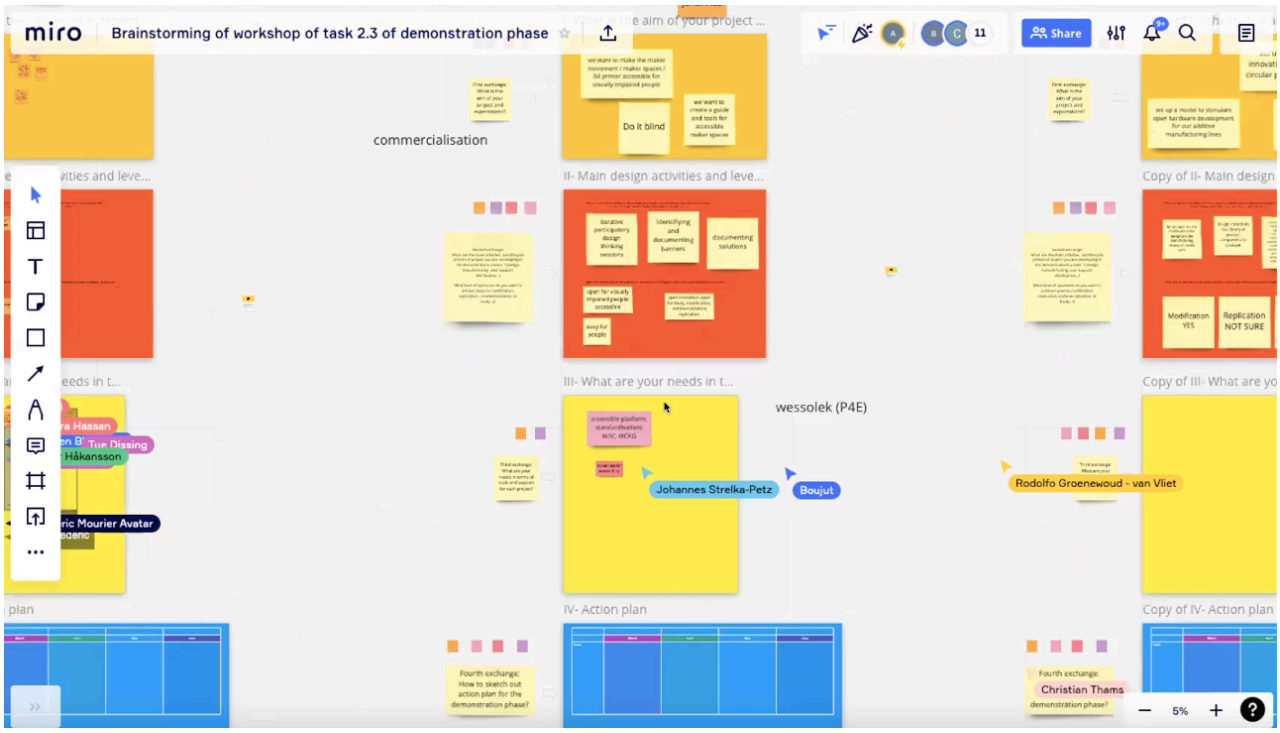


Figure 15: Screenshot that shows the Miro board interface used during workshop

To deepen documentation process with SMEs and OSH founders, we sketched out the monthly meeting with each SME from February to June to cooperate with them in the documentation process and use our guidelines and template of OSH on GitHub or Wikifactory.

The workshops and monthly meetings enabled us to get a better insight into the difficulties, or needs while documenting their designs. Moreover, these monthly meetings were the opportunity to capture the perception and understanding of the description of their practices for the documentation of OSH. The follow up of this demonstration phase included 13 monthly meetings for 6 SMEs between March and May 2022.

4.1.1 Conclusions and lesson learned from the workshops and monthly meetings

Based on the work presented in the above section, we could cluster the participants into 3 groups:

- Those who are in the process of stating what to document (i.e. good understanding of the level of openness they want to achieve according to their business model and OSH development needs)
- Those who already answered that question and question how to do it.
- Those who already published a solid documentation, and consider publish a new version using our methodology and template.

Such categorization led us to question the roots of discrepancies between groups. The first thing noticeable is that the level of openness and the business model were not very well defined for the majority of the projects. We can illustrate that with verbatim from the workshop:

"What level of openness do I want?
- Not entirely decided yet…,
- -Replication?
- Not sure"

If documentation of the open-source project is often perceived as "not a priority", burdensome, time consuming, complex because the lack of modelling tool, communication tools, etc., this usually means that some serious questioning needs to be done, prior to engage in the documentation process.

Participants need to clarify what is expected from the input from external stakeholders, (makers, users, investors, etc.). It mainly means clarification of the business model, often not well defined at first for makers. Business model, or different type of expected rewards other than money, could impact, constrain, limit the openness of the project and therefore directly impact the content and process of documentation. **These questions must be addressed before starting the documentation process**.

From the answers of these questions emerges a strategy of openness and therefore a strategy for documenting the project. The selection of documentation section, the file formats used, the license, ... will be defined as part of high-level strategy. This inquiry brings two main results:

1 - Requirements concerning documentation process derived from a strategy
2 - Documentation should be considered as an integral part of design process

The two illustrations below aim to summarize two different situations we encountered. Figure 16 shows the situation that concerns the lack of anticipation and reflexion, which is the most common situation. Figure 17 shows practical problem encountered. It assumes that the prior strategic questioning is already done. In this second situation the solution is usually easier to find.

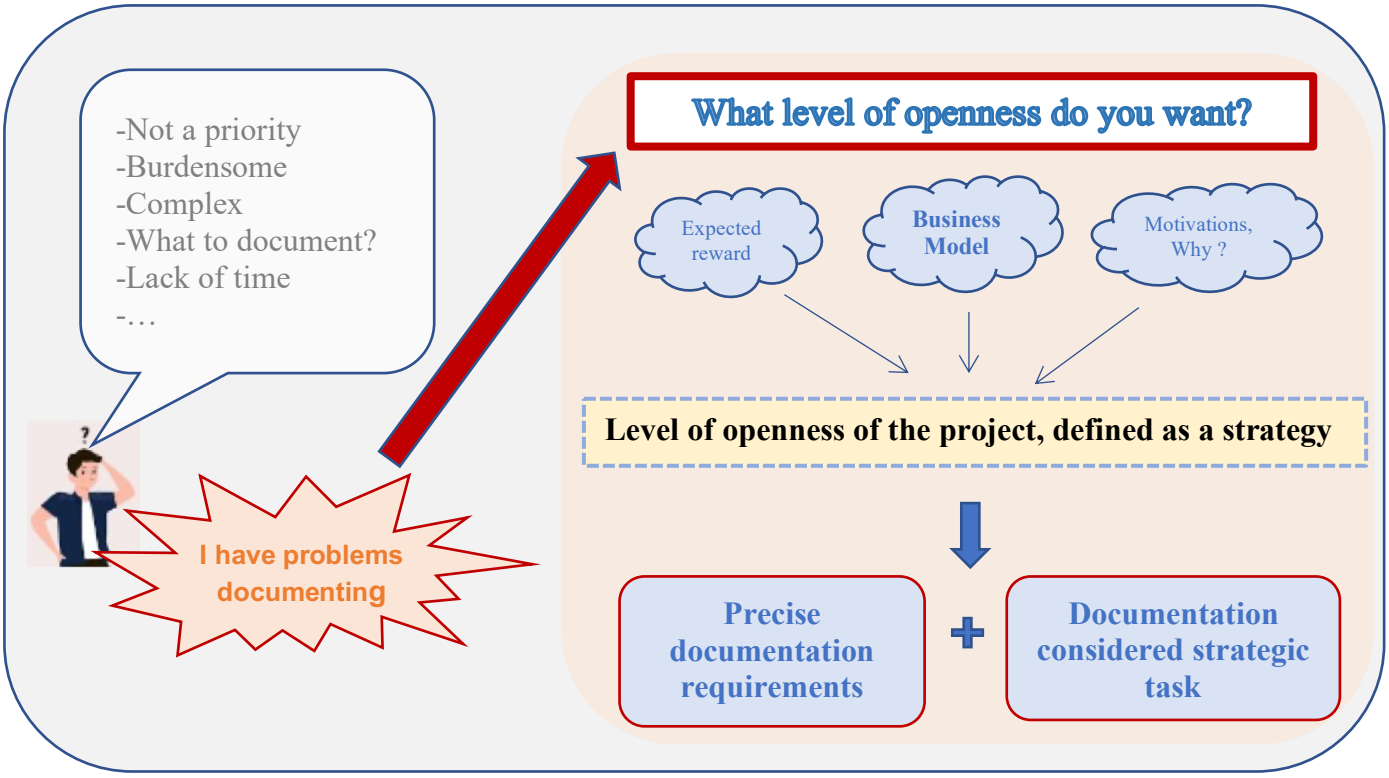


Figure 16 : Lack of prior questioning impairing the documentation process, steps to follow and outcomes

If makers have problem documenting and report these "explanations" they should proceed as follow:

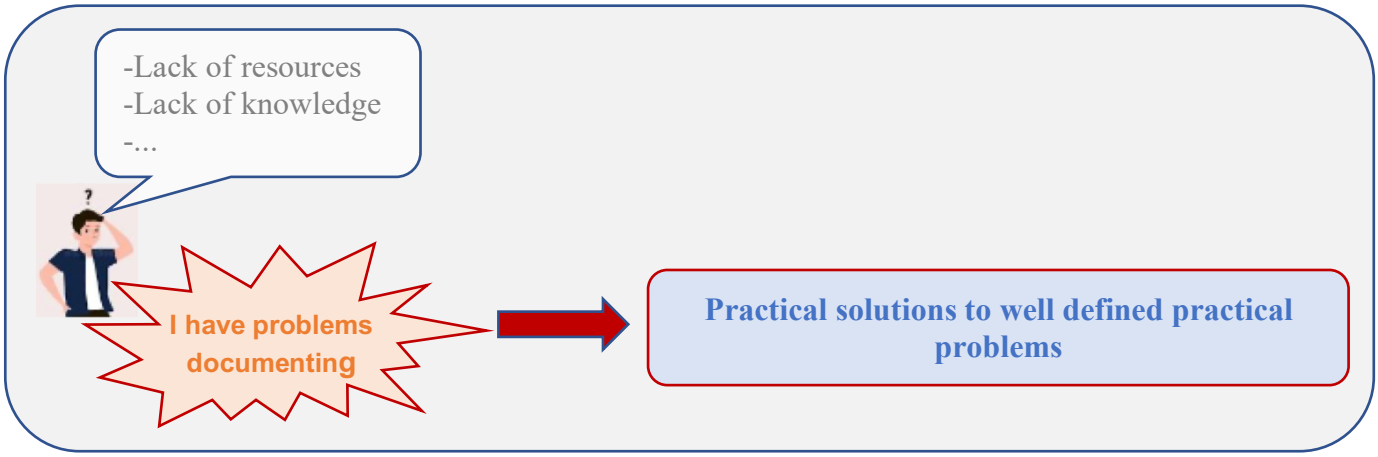


Figure 17:Practical problems impairing the documentation process

As pictured in figure 17, other practical questions could prevent the actors to properly document their projects. It must be clearly identified as real problems, not excuses hiding a lack of documentation strategy. If identified as such, it should therefore be addressed as practical problems with practical solutions. Some examples of usual problem mentioned:

- Lack of time and resource: *"We need time or recruitment to fill those tools. But currently we miss time"*
- Lack of knowledge about right tools (Github, sphinx, Wikifactory, Hedgedoc, GitHub Pages, Discord, ...)

4.2 Qualitative analysis of well documented projects

Whereas the operational evaluation with partners concentrated on the assessment of the correctness of our guidelines, in this section, we try to evaluate the completeness. Since makers are often reluctant to document OSH designs, we worked out indicators that demonstrates how the quality of a documentation, especially its completeness, impacts the quality of the community (size, dynamics, etc.). This work is based on the qualitative analysis of 3 projects enrolled in the demonstration phase and 6 well-documented OSH web-based projects.
The three SMEs projects with good advancement in the documentation process are:

- Stroem
- Manyone
- Anavi

The well-documented web-based OSH projects are:

- NASA JPL
- Poppy 3D printed robot
- SatNOGS Rotators V3
- Farmbot
- Makair ventilator
- BCN3D MOVED

4.2.1 Checking the completeness of our documentation

This review aimed to compare the content of each project with the content of guidelines to have a measure of the coverage of each project. We mainly looked for the types of documents available in comparison with the requirements of our guidelines. We also looked for existing sections of documentation that were potentially not included in our guidelines. The method was to systematically scan each project and look for the content of documentation.

Results: Following the comparison between guidelines and sections observed in projects documentation, we found that the most frequent documentation sections are (see figure 18 below):

- Introduction (mentioned by 9 project)
- Structural model of software, assembly, use (mentioned by 8 project)
- Structural model of mechanic, manufacturing (mentioned by 7project)
- Structural model of electronic (mentioned by 5project)

The less frequent documentation sections are:

- Specification (mentioned by 4 project)
- Behavioural model (mentioned by 3 project)
- Functional model, maintenance and disposal (mentioned by 1 project)

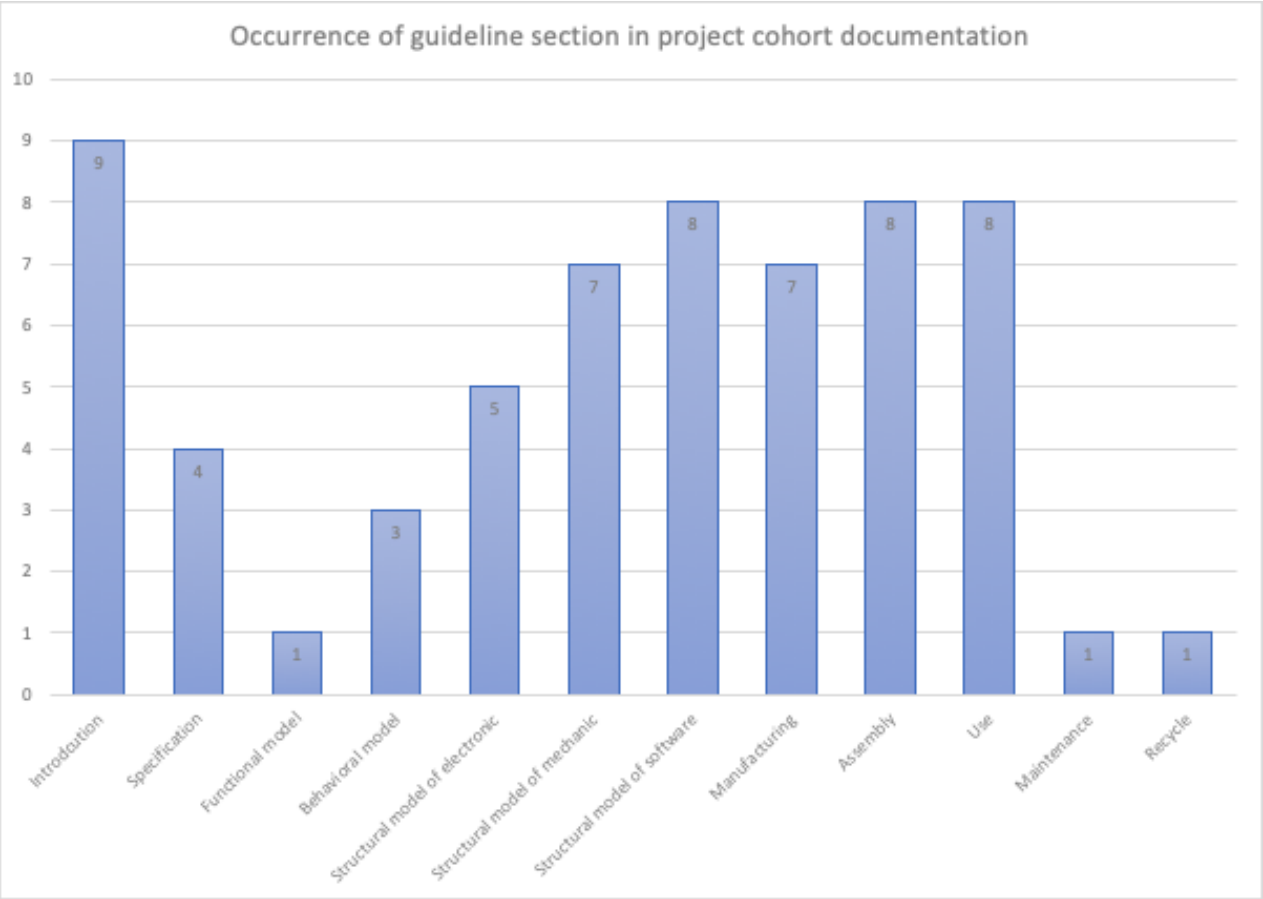


Figure 18: Occurrence of guidelines section in project cohort documentation

The *introduction* is very often present as it identifies the goals, motivation, history of the project, structure of the project, and contribution process. It is the most common section in our review.

Structural model of the software, *Use*, and *Assembly* are also well represented sections with 8 occurrences, followed by *Structural model of mechanics* and *Manufacturing* with 7 occurrences. Although less present - with 5 occurrences - the structural model of electronics appears to also be a strong section of documentation practices.

The frequency of documentation in each project shows that the pilots and well-documented projects have completed a maximum of ten (over 12) sections of our guidelines. In conclusion, this frequency demonstrates that all of our guidelines' sections have been addressed at least once and that no section was missing. No other section in existing documentations have been discovered.

4.2.2 Review of the relationship between documentation completeness and community

This section aims to consider the effect of a complete documentation on the community collaboration dynamic.

The methodology for evaluating the dynamics of collaboration was based on the review of the platform of documentation of each project with the following metrics:

- Number of contributors
- Number of Forks
- Number of open issues
- Name of site and repository of documentation
- Number of stars
- Number of watches
- Number of followers
- Number of active persons
- Number of discussions
    - Number of messages
    - Number of persons

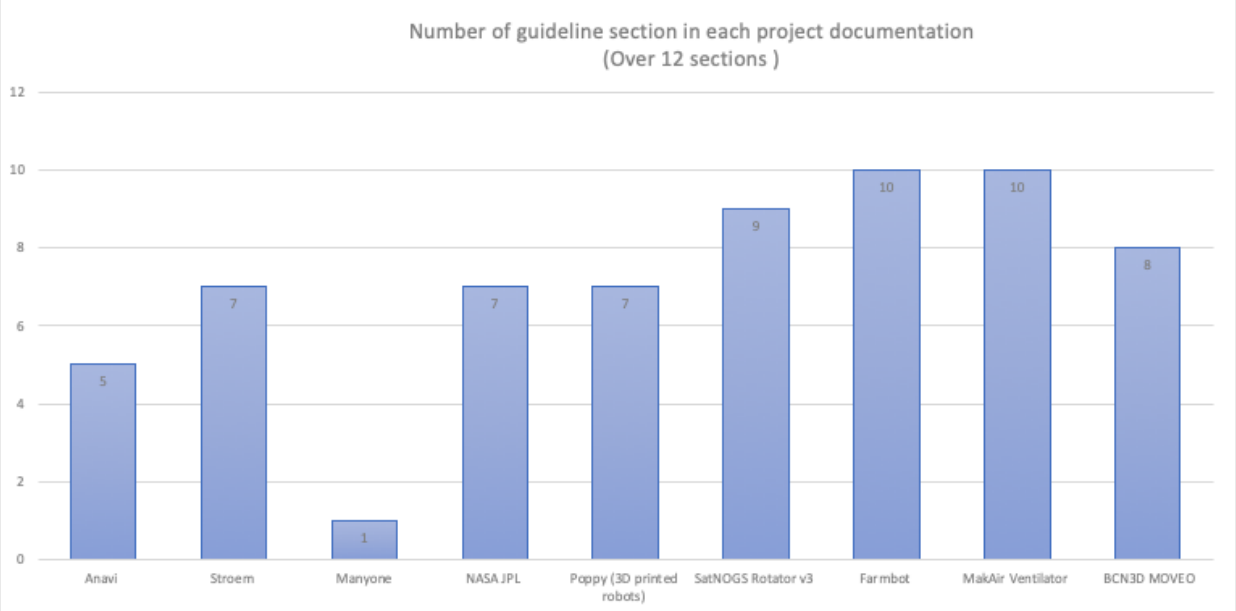


Figure 19: Number of guidelines section in each project's documentation

On one hand, as shown in figure 19, the most complete projects according to the provided sections of documentation are:

- Farmbot and Makair (10 sections of documentation)
- SATNOGS rotator v3 (9 sections of documentation)
- BCN3D MOVEO (8 sections of documentation)
- NASA JPL, POPPY 3D printed robot and Stroem (7 sections of documentation)

Additionally, we considered the dynamic of the community in the nine mentioned projects in figure 20. We have selected three markers of dynamism:

- Number of forks
- Number of contributors
- Number of open issues

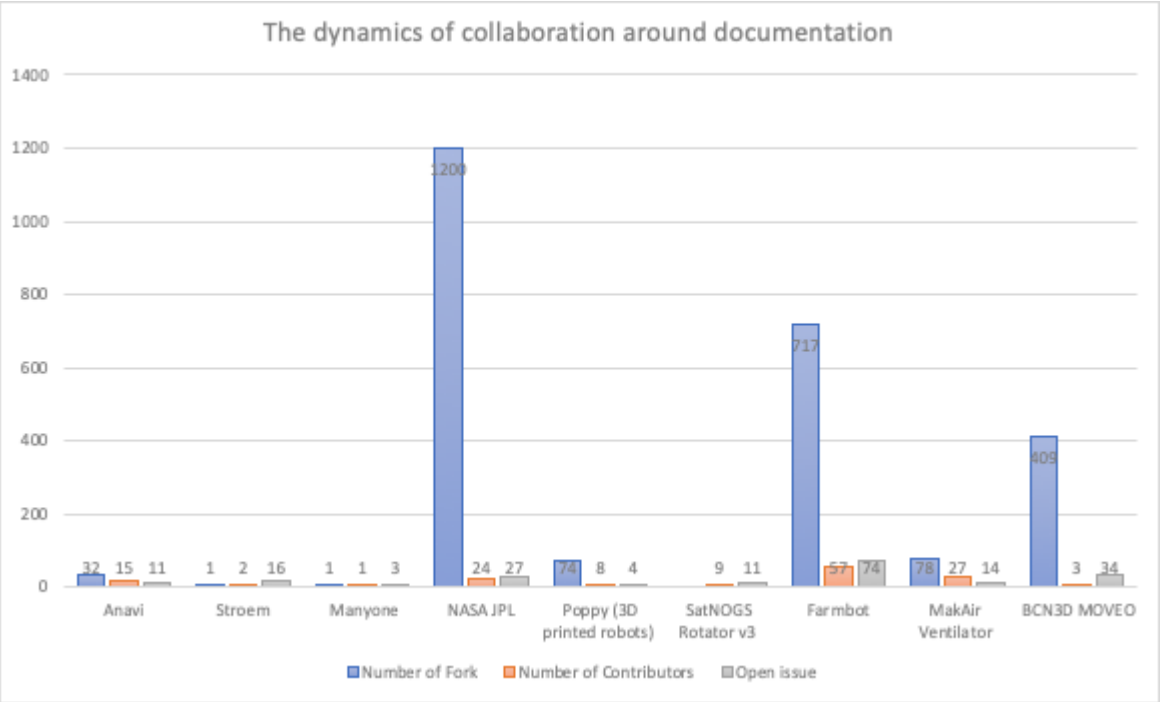


Figure 20: Depiction of the dynamic of community around projects

While our sample size is fairly small to draw sound conclusions, we could state some educated results. The analysis indicates that the more complete projects are from a documentation point of view, the more dynamic the community tends to be. A complete documentation tends to increases the number of contributors, fork, and open issues in the OSH projects. The age of the projects should be considered. Makair started in 2020 when NASA JPL first issues were raised in 2018, and Farmbot first GitHub page on 2014.

Although these first outcomes tend to indicate that the completeness and quality of documentation positively impacts the community and the dynamic of collaboration, it should be verified with additional research. Other data can be considered, for example stars, follower, repositories, issues tracking information, specifically study informal or formal teams collaborating around the project, etc.

In other words, a well-structured documentation comporting relevant sections could be considered essential to enable design reuse, but further research is needed to demonstrate it and specify to what extend this implication is true and what are the other factors.

However, we did not try to prove that solid dynamics around project collaboration meant that the project was well documented. Neither did we correlate the implication of poor documentation practices over collaboration dynamism. These questions should be addressed in future works.

4.2.3 Initial analysis about the current (good and) bad practices of makers documenting OSH project.

Our observations stressed that functional and behavioural models were often not documented. The documentation is usually focusing on the final picture of the detailed design and how to build it. All specification and preliminary design activities including functional analysis,

architecture design, trade-off analysis are often not documented because makers are motivated by the production of the idea they have in mind. The projects are composed of relatively small teams of makers that are enthusiast to rapidly deliver new product to the world. The lack of a contractual supplier/client relationship makes less obvious the need for documenting upstream and downstream new product development activities. Indeed, OSH development is more a hobby that adopts a trial and error iterative design process without customers' needs to satisfy and consequently no costs, delays and risks to control. However, outside this makers sphere, such a design approach happens to be very limited when the project grows and ambition to be widely disseminated and replicated, especially to establish a company-community collaboration, without a well-defined documentation of the OSH product development process.

5 Summary and outlook

This work addresses the need of guidelines for documentation of OSH as defined in the proposal of the OPENNEXT project, work package 2, task 2.3. It depicts the content of the guidelines, the associated template, implementation and guidance facilitating its usage according to user's activity models. Today the open-source development practices are suffering from a lack of structure and guidelines and consequently it is very hard to find good quality information from existing designs. In our study we found that the most documented aspects were on the one hand the contextual and motivation aspects, and, on the other hand the data for replication (manufacturing files and bills of material). User manuals are also found in some more mature projects that distribute kits for example. The missing point is often the information required for modifying and deeply studying the design of the product. This tends to hinder the possibility to reuse the designs and therefore to contribute to a cumulative effort that is the essence of open-source and community-based development.

Our overall contribution aims at establishing best practices in the documentation of open-source hardware products that cover the whole design process especially emphasizing the dimension of modifying and reusing existing designs. The guidelines we designed stand as minimum viable documentation as a response to current poor documentation practices in the development of OSH. The complete template is available on the well-known GitHub collaborative platform. The Wikifactory implementation of the guidelines enables the selection of documentation sections to adapt to users' needs. In order to construct the guidelines, we have followed a structured and systematic approach that was inspired by the industrial product development practices.

The demonstration phase focused on follow up with SMEs. In addition to guide them through documentation process, templates and platforms, the workshops and meetings highlighted recurrent misconception and blockage about documentation and enabled the identification of root causes. The frequent reluctance to document open designs is mainly due to a lack of strategic questioning about the openness of the project and, more generally, and ill-defined business case. Therefore, our work resulted into two main conclusions:

- Firstly, a strategic vision of the openness of the project, associated with a business model reflection enforces requirements about documentation structure and content. This allows the team to design the documentation contents according to the openness and business model strategies.
- Secondly, we recognize the strategic importance of documentation and propose to consider it as part of the product design process itself. This allows document related tasks and activities to be planed and resources to be allocated to these aspects.

Early findings depict that a complete and correct documentation seems correlated to a highly dynamic community and a greater number of reuse of designs by other makers. Successful projects that are often supported by a good documentation and should motivate makers with new projects to implement the minimum viable documentation. These learnings give good indications of the relevance of the documentation process for SMEs and OSH communities we developed. Our work helped promoting good documentation practices for SMEs and makers and allow to clarify the requirements for actually providing designs that are replicable and modifiable and therefore compliant with the OSHWA standard.

References

Cooper, R. G. and Kleinschmidt, E. J. ,1988, “Stage Gate Systems for New Product Success,” *Mark. Manag.*, vol. 20, no. 1–4, pp. 20–29,

Dorst, K. and Vermaas, P. E. , 2005, “John Gero’s Function-Behaviour-Structure model of designing: a critical analysis,” *Res. Eng. Des.*, vol. 16, no. 1–2, pp. 17–26,

Ezoji, A., Boujut, J.F. and Pinquié, R., 2021, “Requirements for design reuse in open-source hardware: a state of the art”. Procedia CIRP, 100, pp.792-797.