Prototyping a sustainable GUI for an IoMusT project: A reflection on adopting sustainable practices

*

1st Elisa Silene Panozzo
Dept. of Information Engineering
University of Padova
Padova, Italy
elisasilene.panozzo@studenti.unipd.it

2nd Mela Bettega
FCSH
NOVA University Lisbon
Lisbon, Portugal
mela.bettega@protonmail.com

3rd Antonio Rodà
Centro di Sonologia Computazionale
University of Padova
Padova, Italy
roda@dei.unipd.it

4th Nicolo' Merendino
Centro di Sonologia Computazionale
University of Padova
Padova, Italy
chihauccisoilconte@gmail.com

***Abstract*—This paper presents a reflection based on the development of a Graphical User Interface (GUI) for an Internet of Musical Things (IoMusT) project designed in accordance with a set of sustainable practices. The case study in object illustrates the feasibility and benefits of incorporating sustainability into digital interface design. Through this exercise, we demonstrate the potential for creating environmentally and socially sustainable digital solutions and provide a concrete example that can inspire future work in this area.**

***Index Terms*—Sustainability, HCI, Design, IoMusT, IoT, FLOSS**

I. INTRODUCTION

The Internet of Musical Things (IoMusT) is an emergent field within the broader Internet of Things (IoT) framework, dedicated to integrating musical devices and systems into interconnected networks. [72] IoMusT offers innovative opportunities for real-time collaboration and enhanced musical expression [75].

However, like many areas of computing [28], IoT and IoMusT projects raise important questions regarding sustainability—both in terms of environmental impact and long-term usability [41].

The project at the center of this study is an extension of "Below 58 BPM" a pre-existing IoMusT system. [45]

This initiative is particularly meaningful, as it aims to support a singer who has overcome a challenging medical condition, enabling her to continue her artistic career. By applying sustainable design principles to this project, we highlight the intersection of sustainability, technological development, and human well-being, demonstrating how thoughtful design can positively impact individuals and the environment.

More specifically, this paper focuses on the development of a Graphical User Interface (GUI) for the IoMusT system supporting this singer. The primary goal of the empirical work was to create a GUI that adheres to well-established sustainable design principles [40]–[43] . By sharing this process and our reflections, we aim to showcase the potential for creating environmentally and socially sustainable digital solutions, offering a concrete example to inspire future work in this area.

The reflections presented in this paper are based on the experiences of a young developer (Author 1) who built the GUI according to a set of sustainable practices. These reflections stem from a diary kept during the development process and a final interview about her overall experience. Through the analysis of this material, we were able to examine the elements she integrated seamlessly into her practice and those she found more challenging. As a recent graduate, Author 1's perspective is valuable because the challenges she encountered reveal specific gaps in sustainable design education within academic curricula.

By sharing our insights and lessons learned, we aim to contribute to the collective knowledge on sustainable design and encourage more researchers and practitioners to embrace these practices.

The rest of the paper is structured as follows: section II presents an overview of relevant literature on IoT, IoMusT, and GUIs in relation to sustainability; Section III details the research objectives and methodology; Section IV describes the process of designing a Sustainable GUI for a pre-existing

Identify applicable funding agency here. If none, delete this.

IoMusT system. Section V presents the main results, which are further elaborated in the discussion and conclusion found in Section VI.

II. Background

Several research studies have explored the sustainable development of interfaces for IoT-relevant projects from various perspectives and disciplines. In the following paragraphs, we provide a brief overview of how recent research has intersected the realms of HCI, IoT systems, and musical projects with the topic of sustainability.

A. IoT, IoMusT and GUIs

The **Internet of Things** (**IoT**) represents a paradigm shift by connecting everyday devices to the Internet, allowing them to communicate, collect, and exchange data [7], [64], [65]. This interconnected network encompasses various applications, from smart homes and wearable devices to industrial systems and urban infrastructure, significantly enhancing efficiency, automation, and user experiences [4]. Building on the IoT framework, the **Internet of Musical Things** (**IoMusT**) [72] emerges as a specialized subset dedicated to the musical domain. IoMusT involves the integration of musical instruments, devices, and wearables into a connected ecosystem, facilitating innovative forms of musical expression and interaction. This convergence of IoT and music technology transforms traditional music practices and enables real-time collaboration and adaptive performances, thereby expanding creative possibilities for musicians and composers in unprecedented ways.

As reported in scientific literature, the use of Internet of Musical Things interfaces can benefit artistic expression by facilitating real-time control, musical parameters' visualization, and more, enhancing the user experience and enabling more expressive performances [71], [75]. **Graphical User Interfaces** are essential components in the processes of controlling and interacting with IoMusT systems, enabling musicians to manipulate sound parameters, effects, and instrument settings through intuitive visual interfaces. For example, GUIs on tablets and smartphones allow musicians to adjust these parameters using touch gestures, making the creation and modulation of music more interactive and user-friendly [13], [14]. Additionally, augmented reality (AR) and virtual reality (VR) technologies have been integrated into IoMusT applications to create immersive environments where musicians can interact with virtual instruments and visual feedback in a three-dimensional space. [10], [67], [76]. Advancements in visual interfaces not only enhance the usability of musical devices but also unlock new opportunities for collaborative performances and educational tools, ultimately expanding the possibilities of traditional music-making.

B. Tools for GUIs development

The number of tools for developing GUIs has increased in recent years. Many Computer-Aided Design (CAD) tools that facilitate the development of visual interfaces are available, and the research community has produced several studies to test and evaluate them [32]. Many UX/UI manuals explicitly suggest the use of popular CAD software and we will provide a brief overview of the most common ones. For vector graphics, powerful tools such as Adobe Illustrator are widely used by designers to create detailed and scalable graphics [25]. With advancements in technology, the industry has introduced more sophisticated tools to help designers create interactive visual interfaces. Tools like Figma have gained popularity for their capability to design interactive elements [15]. Additionally, numerous JavaScript libraries, such as D3.js [24] and Three.js [22], enable developers to create interactive and dynamic experiences on the web. Moreover, drawing from the gaming industry, engines like Unity [31] and Unreal [60] provide robust platforms to develop visual experiences that are highly engaging and immersive for users [70]. These advancements in tools and technologies have significantly enhanced the capabilities of designers and developers in crafting compelling visual interfaces at many layers, from 2D menus to 3D and VR projects.

C. Sustainable IoT

The discourse on IoT and sustainability has largely centered on environmental sustainability, with considerable emphasis placed on advancing a green transition for IoT technology [11] [3]. Given the significant environmental footprint of IoT and networked computational systems [48], researchers scrutinized the large-scale production of these electronic devices. Over time, studies have delved into the environmental impacts associated with the energy consumption of IoT, focusing on optimizing data streaming energy use, enhancing computational efficiency [2], and developing self-sustaining and more efficient hardware [62]. Additional research has addressed environmentally sustainable strategies targeting specific components, such as minimizing the impact of individual protocols like Bluetooth [69] or optimizing task scheduling to conserve energy [59].

In recent developments, Mahmoud and colleagues introduced the concept of "Green IoT" (GIoT) [1]. Their comprehensive review of sustainable IoT development identifies various enablers, architectures, technologies, energy models, and categories essential for advancing green IoT.

Similarly, in their paper "Green IoT: An Investigation on Energy Saving Practices for 2020 and Beyond", Arshad et al. outline **six key elements for transitioning to Green IoT** [3] In addition to this taxonomy, research on sustainable design has extensively highlighted the importance of tools and devices' longevity in reducing environmental impact, with e-waste being a significant issue in digital technology. Recently, the IoT debate has included this topic, with Lechelt et al. [36] advocating for updating, recycling, and upcycling IoT devices under the "end-of-life" concept, highlighting how the embedded value systems within an IoT object can influence its lifecycle by encouraging individuals to retain, repurpose, or recycle the object, or rethink its use after it has reached the end of its functional or operational life.

D. Sustainable HCI, FLOSS, and NIME

Nearly two decades ago, academics operating in the field of Human-Computer Interaction (HCI) began to engage with specific projects and reflections targeting sustainability, giving birth to the sub-field of Sustainable HCI (SHCI) [23], [28], [37].

SHCI has increasingly adopted a holistic approach, extending the concept of sustainability to include social and economic aspects [28]. In 2007, Mankoff et al. [**?**] proposed a simple but effective classification to unpack how digital tools support sustainability. The authors identified two distinct ways in which digital technology can support environmental sustainability: *in* design and *through* design. *Sustainability in design* pertains to the material aspects of a product, embedding sustainable features such as reduced energy consumption. *Sustainability through design* focuses on the potential of an artifact to encourage environmentally friendly behaviors.

For instance, the existing literature provides guidelines to design digital tools with reduced energy consumption [12], [53], [61], [66] can be considered a contribution to sustainability *in* design concerning tools' environmental impact.

Similarly, the guidelines that aim at improving accessibility contribute to the sustainability *in* design concerning the inclusion of categories that are typically less privileged (and doing so oftentimes improves the design for other categories of people as well) [19]

Companies like Microsoft [1], Facebook [2], and Google [3] are increasingly committing to improving products' accessibility and environmental sustainability relying on specific guidelines. Such guidelines involve, among other things, the use of dark mode, high contrast colors, and dimensions of texts. However, the fact that these companies can achieve a good standard in terms of accessibility and on some indicators of environmental sustainability does not imply that they are committing to other aspects of sustainability such as supporting democratic values, social justice, and a deeper engagement with environmental sustainability rooted in improving processes' efficiency and promoting hardware lifespan.

In this regard, Free Libre and Open Source Software is typically recognized as a better performing model [6], [8] and several scholars deem it is worth dedicating a specific effort to promote its use among the population [18], [35], [44], [57], [58].

Free Libre Open Source Software (FLOSS) [73] - or just OSS in some definitions [27] - refers to software released under a specific type of license that makes its source code legally accessible to end users. Software is defined as Open Source if its source code is available at no additional cost, allowing users to view and modify the code according to their needs. Additionally, in Open Source projects the source code can be reused in new software, enabling anyone to use it to create and distribute new programs.

[1] https://www.microsoft.com/en-us/accessibility

[2] https://www.facebook.com/help/273947702950567

[3] https://www.google.com/accessibility/

One of the main distinguishing features of FLOSS is the fact of relying almost entirely on the voluntary commitment of the developer's community [17], [56] (although some projects are coordinated by commercial entities [29]) This aspect keeps the development and testing costs low (Total Cost of Ownership, or TCO, approximates to zero), making such software accessible to less privileged social classes.

For instance, since we mentioned CAD software in relation to GUI development, the best-known commercial CAD applications require the purchase of expensive licenses, and their outcome can be subject to obsolescence due to the choices of the commercial company that developed them. Conversely, FLOSS CAD software such as Inkscape provides an alternative that is free of cost and is potentially more reliable on the long run as it is not dependent on the single-sided decision of a private entity but rather from the collaboration of a community of developers.

Sustainability was also subject to reflection in the field of music projects. In particular, the New Interface for Musical Expression (NIME) community, made a significant effort in reflecting on the environmental impact of the works presented by the community. In fact, in the past years, notable attention has been dedicated to the prevention of obsolescence of musical interfaces. Such a topic has been investigated through specific case studies such as the T-tree and others [33], [50] as well as through several positions papers [47]. Notably, the term "disposable instrument" [16] has been created in NIME to define interfaces that are used only once before remaining unused. Furthermore, in NIME 2024, nine sustainability strategies have been discussed to provide the community with specific guidelines to develop DMIs more sustainably [46].

III. Methodology

The main objective of this paper is to reflect on the implications that designing a GUI following sustainability principles can have for designers. These reflections are based on a specific case study in which we designed the Mock-up of a Graphical User Interface to complement a previously developed IoMusT device.

The designer who implemented the GUI (Author 1) has completed a bachelor's degree in informatics and is familiar with coding and application programming. However, she was not familiar with the sustainable design principles we applied, including the FLOSS tools that we decided to use.

To provide some relevant reflections on the implications of following sustainable design predicaments while designing a GUI, we proceeded as follows.

1) We scrutinized a series of research works and guidelines (reported in the background section) and funneled them into a series of specific characteristics and actions that the design of our interface should embed to align with a sustainability-oriented approach.

2) Author 1 developed the GUI Mock-up having particular care in following the requirements we pinpointed. During the whole process, she compiled a diary to keep track of her

overall experience, including the informal conversations that occurred during the various design activities.

3) After the GUI Mock-up has been delivered to the artist, Author 4 interviewed Author 1 in regards to her experience as a designer engaging for the first time with specific sustainability guidelines.

4) Since all the people involved in the project were Italian, it has been necessary to translate in English both the diary and the interview. Afterwards, the transcriptions were analyzed through an inductive thematic analysis [26], [55] that led to the definition of three main themes that will be reported in the results section. The transcript and the thematic analyses are available in this public repository [4].

In addition to this process, we briefly interviewed the artist for whom the IoMusT and the GUI were created to validate the quality of the outcome.

IV. Case study

To provide a bit more context, this section begins by briefly introducing the IoMusT device for whom we developed the GUI. Afterward, we will describe in detail the design process that led to the development of the GUI mock-up.

A. The IoMusT system: "Below58BPM"

"Below 58BPM" is an IoMusT system that has been co-designed in collaboration with Eleonora Amianto. Eleonora is an opera singer who suffered from a carotid aneurysm and needed to find new strategies to be able to pursue her artistic career despite her difficult health condition.

"Below 58BPM" comprises a wearable element and a client software. The design process and the device are presented in detail in [45]. Here we present the main elements. The wearable element is a collar with embedded sensors and actuators that connect to a client software via an esp32 microcontroller.

The system monitors a critical biometric parameter (heart rate) in real-time; in case the heart rate exceeds the critical threshold of 58 Beats Per Minute, the system:

- 1) warns the singer through haptic feedback;
- 2) apply sound distortion to the singer's voice; the singer can manipulate these sound effects through four knobs positioned on the collar.

The possibility to rely on these sound effects allows the singer to rest without interrupting her performance as she can alternate physically demanding opera singing techniques with less demanding singing techniques and even self-healing breathing sessions.

Lastly, "Below 58BPM" stores the data acquired on an online platform. The resulting database can be used by the singer to evaluate and compare sessions over long periods of time.

[4]https://github.com/chihauccisoilconte/B58BPMGUI_research

B. GUI Design

The main goal of the GUI was allowing the artist to smoothly interact with the database containing her biometrical data, allowing her to compare different performances. In this subsection, we will describe in detail the design activities related to the development of the GUI following sustainable design principles.

After Author 4 performed a brief design session with the singer to define the functions and basic structure of the interface (see hand-drawn mock-up represented in 1) we proceeded to: 1) define the sustainability-oriented characteristics that the interface should have and selecting Open Source software in line with sustainability literature; 2) performing the CAD and coding sessions that led to the actual production of the mock-up.

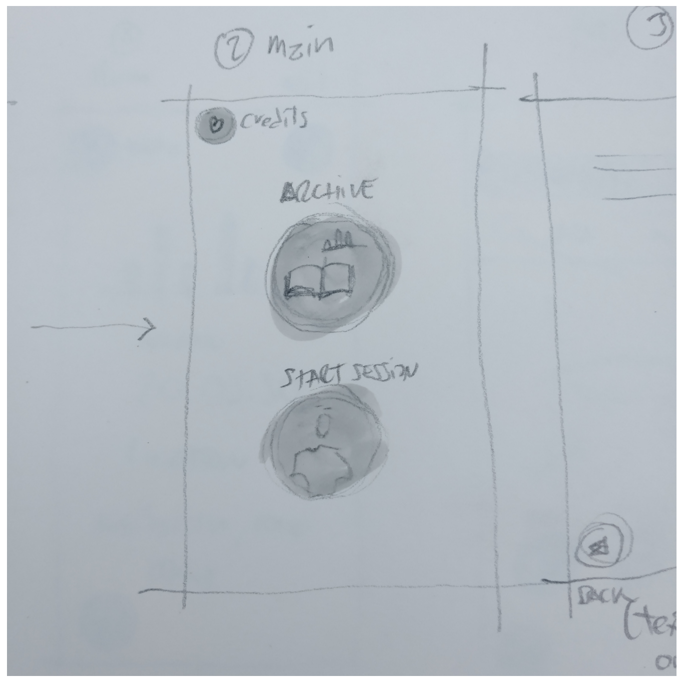

Fig. 1. Picture of the hand-drawn interface mock-up made during the session with the artist

1) Definition of sustainable practices: Since our goal was to design a GUI following sustainability principles, a crucial step of the process has been researching the existing literature and guidelines II and funneling this information into specific design requirements/choices.

a) Graphical elements: Since effective UX/UI design relies not only on context, layout, and interactions but also on typography, font selection and Hierarchy were key aspects of our design process.

The choice of **fonts** can significantly impact usability, particularly for visually impaired users [9]. According to Bootcamp, a source on UX, UI, and design research, **Readability** [77] is a key parameter to consider while selecting a font. Well-readable fonts ensure that text is legible at all sizes and contrast well with the background to reduce eye strain. For this project, we selected Nordique, a sans-serif font known for its readability.

We also considered the importance of **Hierarchy** [54] which is the arrangement of text elements to guide the reader through the content in a structured manner. Specifically, we used different font weights (bold, italic, etc.) and sizes to create a visible hierarchy, facilitating easier reading and comprehension.

Alongside typographic choices other elements such as **buttons, sliders, and menus items** are subject to similar considerations. Therefore we made sure that every visual element was clearly visible and we avoided overcrowding screens with too many elements, reducing visual noise.

b) Dark Mode: Color choice is a crucial factor in the sustainability of a GUI as it affects both elements' visibility on the screen and the screen's energy consumption. Therefore, we dedicated some time to defining which colors to employ; despite some drawbacks pointed out by [52], we chose to implement dark mode in our GUI design. Dark mode is advantageous for energy savings, enhances the visibility of interface elements, and reduces eye strain in low-light environments. However, it may cause discomfort in brightly lit settings. Dark mode doesn't require pure black to achieve energy savings. For example, Google's surface color #121212, while not pure black, consumes only 0.3% more energy than pure black

c) Use of FLOSS: Well-established literature and guidelines II emphasized the importance of using Open Source software to promote environmental and social sustainability, For this reason, we chose several open-source tools for our project:

For designing static graphic elements, we used **Inkscape**[5] (Fig. 3.1), a free vector graphics software. Originating from Gill (the GNOME Illustrator application) and developed by Raph Levien, Inkscape was designed to fully utilize SVG standards [34]. It is a robust, free alternative to commercial vector graphics software and is frequently used in various design projects.

To transform graphical elements into interactive objects and to develop the app for export to Android, we chose **Godot Engine**[6] (Fig. 3.2). Godot Engine is an Open Source 2D and 3D game engine [30]. Despite being free, it is a powerful tool that competes with well-known engines like Unity and Unreal. It supports multiple platforms, including Windows, iOS, Android, and web games [68]. As with other Open Source software, Godot Engine aligns with sustainability by being freely available and less prone to obsolescence.

By using these open-source tools to develop the GUI, we supported sustainable practices and we could benefit from the use of high-quality, versatile software.

2) Performing CAD and coding sessions: During the initial design session with Eleonora, we outlined the basic structure of the GUI. Its two main objectives are: 1) detecting and sending heartbeat data to the platform; 2) allowing Eleonora (or other potential actors such as researchers) to consult and edit these data. Based on these elements, we defined the basic characteristics of the GUI:

- Title page - welcomes the user while the app loads
- Main Menu - works as a gateway to the two main features of the app (record and archive) plus the credits
- Archive page - contains a list of the recordings archived in the past
- Concert page - allows to visualize the recording of a specific performance, to add comments, and to edit adding information such as the location of the concert, etc.
- Start record session - guides the user in the process of recording the data detected during a performance
- Credits page - contains the credits

a) Graphic design with Inkscape: We used Inkscape to create the primary graphic elements of our interface (fig. 2). Inkscape provides essential tools for this task, including Bézier curve drawing, color management, and paint bucket tools.

To ensure the interface was visually appealing and functional, we avoided using pure black (#000000) and pure white (#FFFFFF), as these colors can create harsh contrasts. Instead, we opted for shades of gray to provide a softer appearance. Additionally, we avoided overly saturated and neon colors, which can hinder readability, and included popular color choices for dark mode UI (purple, blue, and salmon pink). Inkscape's advanced color management tools allowed us to easily implement these design choices. We also utilized Inkscape's wide range of filters and extensions throughout the design process. A notable feature was the ability to organize the canvas in separate layers, which streamlined the creation of graphic elements and their subsequent export for use in Godot. The result of this phase was a series of SVG elements (fig. 3).

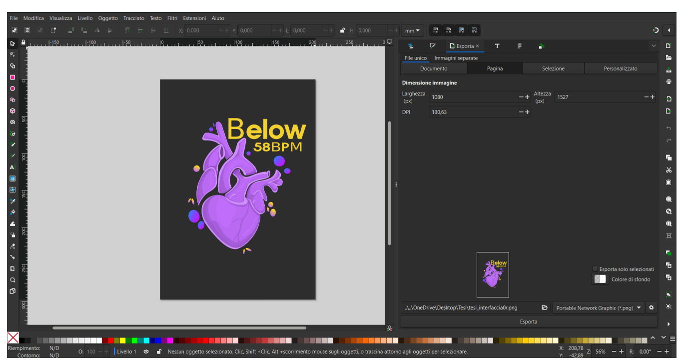

Fig. 2. Screenshot of the project in Inkscape

b) Coding in Godot: To transform the static graphic elements into a functional GUI mock-up, we utilized the Godot Engine (fig. 4). While Godot is often promoted as a tool for 3D game development, it is also well-suited for creating 2D interactive elements, as it was originally designed for this purpose.

We leveraged Godot's features extensively throughout our development process. The GDScript language proved to be

[5]https://inkscape.org/
[6]https://godotengine.org/

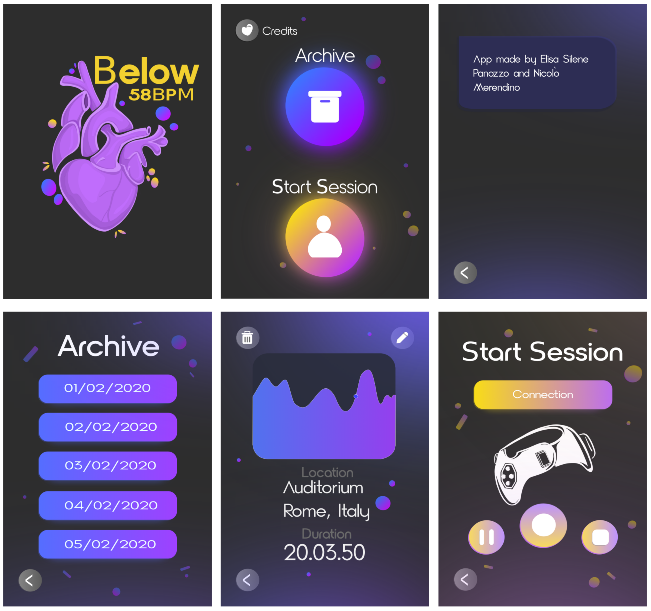

Fig. 3. Static SVGs developed with Inkscape

a powerful tool for adding interactivity and enhancing user engagement without relying on the inspector editor. Here's how we used Godot to achieve our goals:

- **Layout Organization**: We divided the interface into multiple pages, which facilitated a structured and user-friendly design. This segmentation involved creating nodes to connect various aspects of the project and establishing a clear hierarchy, improving organization and navigation within the app.
- **Interactive Elements**: We developed various interactive components, such as buttons, sliders, and other controls, to enrich user interaction and provide a dynamic experience.
- **Performance Optimization**: By leveraging Godot's capabilities, we ensured that the application was rendered efficiently, maintaining high performance and responsiveness throughout.
- **Exporting**: Finally, we successfully exported the application as an .apk file, which can be easily installed on an Android devices from people without specific technical skills.

To conclude, the process resulted in a functional GUI mock-up with fully implemented interactive and visual features. Fig 5 displays the developed pages, and all project outcomes are available in a public repository[7].

V. Results

This section begins by presenting the results of the user feedback session, that highlight the overall high quality of the GUI developed following the sustainable practices we pinpointed. Besides the user's response, the main focus of the work presented in this paper is Author 1's experience designing a GUI with a sustainability-oriented approach. In this section, we will briefly illustrate the user feedback and then we will focus the themes derived from the qualitative analysis performed on the diary and the interview with Author 1.

[7]https://github.com/chihauccisoilconte/B58BPMGUI_research

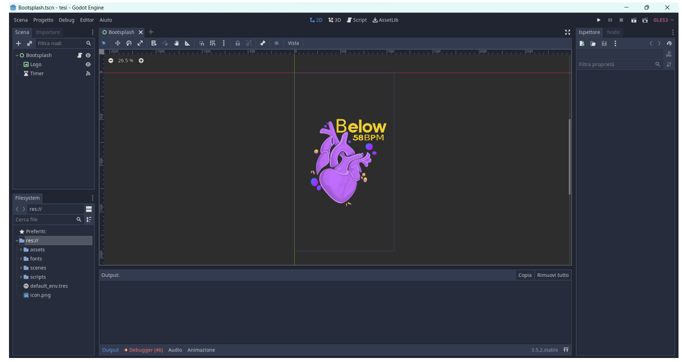

Fig. 4. Screenshot of Godot Engine in use

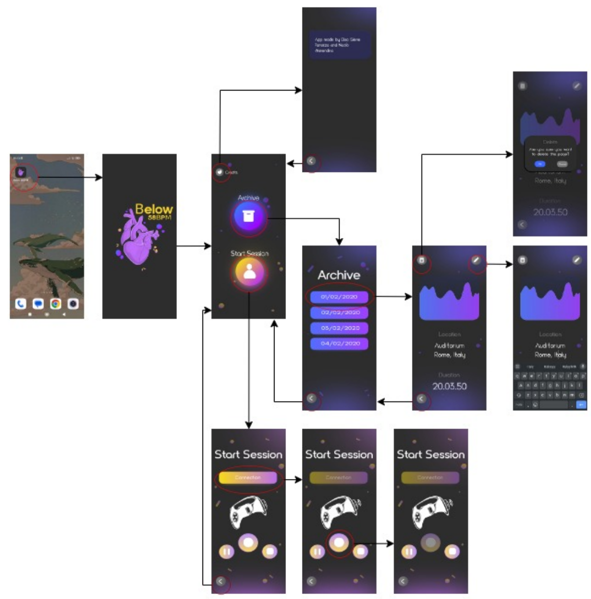

Fig. 5. Interface flowchart

1) User feedback: After developing the Mock-up, we contacted the artist to organize a user testing/feedback session. Since we were operating long distance, we sent her a link containing the APK to install the app on her own phone. During the session, Eleonora installed the app and used the features implemented so far, commenting on her experience. At the end of the session, we asked her some additional feedback.

Overall, the artist provided positive feedback on the GUI. She encountered no issues with the prototype's functionality ("it's like a normal app from the store") and expressed appreciation for some of its aesthetic features. Eleonora also shared some minor negative feedback about the look and the clarity of the ”record” icon; moreover, she was worried about possibility that the GUI could begin to malfunction in the long run.

To address this concern, we explained Eleonora that since the whole project is open-source any person with the right skills could maintain and update the platform for her, if needed. She appreciated both the practical and the political value of the possibility of easily transfer the outcomes of our work to potential future collaborators. To conclude, she foresaw the the potential for using the heartbeat data visualization as a source of artistic content in future projects.

2) Designer's positive reflections on using FLOSS tools (T1): The first theme emerged from the analysis is a general positive impression in using FLOSS (T1). Firstly, the designer noted a series of technical positive features in the FLOSS tools she used.

She claimed, for example, that in comparison to commercial alternatives that she might have used otherwise, Godot presented a series of features that made the process easy as it allowed the programmer to program and visualize the outcomes through the same, easy-to-use interface. In general, she did not perceive the use of FLOSS as a constraint. Besides the satisfaction with the technical aspects of the tools that we used, the designer reported a feeling of emotional satisfaction given by using FLOSS, as she felt being part of a virtuous community that uses tools that are not owned by private companies. In general, she claimed that she will use this software in the future as well as she would recommend them to new users.

3) Lack of FLOSS education (T2): A second theme highlighted in the analysis points to the lack of education in using FLOSS (T2). Author 1 expressed that, at some point, the process was less smooth while developing the Mock-up. Such difficulties were generally given by the fact that she used the software for the first time. Although the software used worked well, finding useful documentation was sometimes difficult (especially concerning Godot). To this end, we point to the fact that evidently, the university where the designer got her education never mentioned even the existence of the software that we used in the development of the GUI Mock-up, underlining the fact that there is a need for more education regarding FLOSS alternatives. Furthermore, when facing specific problems, it was difficult to find tutorials that clearly explained the GUI-related capabilities of Godot, pointing to the fact that some improvements can be made in this sense.

4) Implementing the design features as a challenge (T3): Besides the use of FLOSS, some comments on the design features of the GUI were made. Firstly, author 1 claimed that the implementation of the dark mode was overall simple, but some difficulties emerged when choosing the colors of the interactive elements. In pair with this impression, the designer claimed that choosing the position, shape, and dimension of the elements to make it as accessible as possible was a challenging process that clashed a little with what she had experienced previously in designing a GUI, which is something to be taken into account when starting such a design process.

VI. Discussion

In this section, we further discuss the key elements that emerged in our analysis, connecting our work to the overall debate on sustainability and more specifically to the IoMusT debate.

A. Maturity of FLOSS and education gap

As emerged from the considerations of the developer (T1), the maturity of Free/Libre and Open Source Software (FLOSS) has evolved significantly, establishing itself as a viable alternative to commercial software. Our research contributes to reinforce this point by highlighting the advanced state of development and robustness of many FLOSS projects, which now offer functionality and reliability comparable to proprietary solutions [20], [74], particularly in the design of user interfaces [44]. Our case study showcases the possibility for young professionals to use professional-level software to deliver high-quality outcomes with reasonable effort and within a reasonable time frame. This result complements other research focusing on groups with lower technical skills (and simpler goals) [8], as well as professional groups with limited technical skills supported by mediators [57]. However, our results highlight a critical issue which is the focus of university curricula on proprietary software (T2) which forces young professionals to face extra training if they desire or need to convert to FLOSS options. Although a lot of FLOSS is now reliable and relatively intuitive, there is still a notable gap in its integration within educational institutions, despite its adoption might lead to substantial savings as noted by Pearce [51]

and promote skills development and digital literacy as pointed out by Silva et al. [63].

B. Time management and sustainability

Moreover, an emerging theme from our study is the intersection of time management and sustainability. Sustainable practices within software development necessitate efficient resource allocation and time management, ensuring that projects can thrive and remain viable over the long term. The findings suggest that strategic time management is essential for maintaining the sustainability of sustainable projects, aligning with previous research that underscores the importance of structured time allocation for sustainability (and open-source) initiatives [21]. By addressing these interconnected issues, our research aims to contribute to the broader discourse on adopting work practices to sustainable development [5], [49].

C. IoMusT and sustainability

Additionally, our research presents some reflections on the sustainability of the Internet of Musical Things (IoMusT), emphasizing the potential of FLOSS to drive innovation and sustainable development in this domain. By leveraging

FLOSS, IoMusT projects can benefit from increased collaboration, reduced costs, and enhanced flexibility, which are crucial for the long-term sustainability and evolution of these interconnected musical ecosystems (on music ecosystem see [38], [39]). In general, we claim that the considerations that emerged from our research work take part in the wider debate on sustainability within the field of the IoMusT. Our project answers the call for exploring the categories of sustainable IoMusT launched by Masu et al. [41]

VII. Conclusions and Future Challenges

Our research presents a reflection linked to a set of implications that emerge when applying a sustainability-oriented approach. The outcomes indicate significant room for improvement and development. Firstly, the reflections we propose can serve as a foundational base for further studies involving a larger group of designers, enhancing the breadth and depth of the research. Secondly, the prototype we produced, though promising, remains a basic 2D graphical interface. Future research should delve deeper into the tools utilized in this study to explore their full potential and capabilities, thereby advancing the state of sustainable design practices. In conclusion, whether it's democratizing the use of certain tools, respecting and including minorities in the design and purpose of a product, using affordable and easily accessible materials for ICT technologies, or minimizing waste to respect the environment, these are all topics that should become integral to the HCI community's modus operandi, keeping pace with societal changes and not falling behind, especially on environmental issues. By presenting a case study that aims to incorporate all these concepts and straddling the artistic side of the STEAM community and technological innovation in HCI we hope that such consideration could help the community in moving forward into an ecological transition.

Acknowledgment

This work was financially supported by the Italian Ministry of University and Research, PON (Programma Operativo Nazionale) scholarship n DOT1487343-5, as part of project DM 1061-2021 PON RI 2014-2020- React-EU. A special thanks go to Eleonora Amianto, who put time and effort into making this research possible, and to Raul Masu, who provided very useful insights in different stages of the work presented. Lastly, a big amount of gratitude goes to all the developers who volunteered to make the FLOSS used in this paper, and to the no-longer-existing institution named STEIM (long live STEIM!)

References

[1] Mahmoud A Albreem, Abdul Manan Sheikh, Mohammed H Alsharif, Muzammil Jusoh, and Mohd Najib Mohd Yasin. Green internet of things (giot): applications, practices, awareness, and challenges. *IEEE Access*, 9:38833–38858, 2021.

[2] Mohammed H Alsharif, Anabi Hilary Kelechi, Sunghwan Kim, Imran Khan, Jeong Kim, and Jin Hong Kim. Notice of retraction: Enabling hardware green internet of things: A review of substantial issues. *IEEE Access*, pages 1–1, 2019.

[3] Rushan Arshad, Saman Zahoor, Munam Ali Shah, Abdul Wahid, and Hongnian Yu. Green iot: An investigation on energy saving practices for 2020 and beyond. *IEEE Access*, 5:15667–15681, 2017.

[4] Luigi Atzori, Antonio Iera, and Giacomo Morabito. Siot: Giving a social structure to the internet of things. *IEEE communications letters*, 15(11):1193–1195, 2011.

[5] Rupert J Baumgartner and Romana Rauter. Strategic perspectives of corporate sustainability management to develop a sustainable organization. *Journal of Cleaner Production*, 140:81–92, 2017.

[6] Yochai Benkler and Helen Nissenbaum. Commons-based peer production and virtue. *Journal of political philosophy*, 14(4), 2006.

[7] Dan-Radu Berte. Defining the iot. In *Proceedings of the international conference on business excellence*, volume 12, pages 118–128, 2018.

[8] Mela Bettega, Raul Masu, Nicolai Brodersen Hansen, and Maurizio Teli. Off-the-shelf digital tools as a resource to nurture the commons. In *Proceedings of the Participatory Design Conference 2022-Volume 1*, pages 133–146, 2022.

[9] Nanditha G Bharadwaj, Nanditha Pai, S Mythri, Netra S Soraganvi, Deeksha H Kulal, and M L J Shruthi. Design of a novel smartolight-pen and user interface screen enabled with url detection. In *2022 IEEE North Karnataka Subsection Flagship International Conference (NKCon)*. IEEE, November 2022.

[10] Alberto Boem and Luca Turchet. Selection as tapping: An evaluation of 3d input techniques for timing tasks in musical virtual reality. *International Journal of Human-Computer Studies*, 185:103231, 2024.

[11] David Bol, Guerric de Streel, and Denis Flandre. Can we connect trillions of iot sensors in a sustainable way? a technology/circuit perspective (invited). In *2015 IEEE SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S)*, pages 1–3, 2015.

[12] Valentine Boyev. Dark ui design – 11 tips for dark mode design.

[13] Jamie Bullock, Daniel Beattie, and Jerome Turner. Integra live: a new graphical user interface for live electronic music. In *NIME*, pages 387–392, 2011.

[14] Jamie Bullock and Lamberto Coccioli. Towards a humane graphical user interface for live electronic music. In *NIME*, pages 266–267, 2009.

[15] Dario Calonaci. *Designing User Interfaces: Exploring User Interfaces, UI Elements, Design Prototypes and the Figma UI Design Tool (English Edition)*. BPB Publications, 2021.

[16] Joanne Cannon and Stuart Favilla. The investment of play: expression and affordances in digital musical instrument design. In *ICMC*, 2012.

[17] Andrea Capiluppi and Thomas Knowles. Software engineering in practice: Design and architectures of floss systems. In *IFIP International Conference on Open Source Systems*, pages 34–46. Springer, 2009.

[18] Leandro Costalonga, Daniel Hora, Marcelo Pimenta, and Marcelo Wanderley. The ragpicking dmi design: The case for green computer music. In *Proceedings of the 10th International Conference on Digital and Interactive Arts*, ARTECH '21, New York, NY, USA, 2022. Association for Computing Machinery.

[19] Sasha Costanza-Chock. Design justice: Towards an intersectional feminist framework for design theory and practice. *Proceedings of the Design Research Society*, 2018.

[20] Kevin Crowston, Kangning Wei, James Howison, and Andrea Wiggins. Free/libre open-source software development: What we know and what we do not know. *ACM Computing Surveys*, 44(2):1–35, March 2008.

[21] Kevin Crowston, Kangning Wei, James Howison, and Andrea Wiggins. Free/libre open-source software development: What we know and what we do not know. *ACM Computing Surveys (CSUR)*, 44(2):1–35, 2008.

[22] Jos Dirksen. *Learn Three. js: Program 3D animations and visualizations for the web with JavaScript and WebGL*. Packt Publishing Ltd, 2023.

[23] Carl DiSalvo, Phoebe Sengers, and Hrönn Brynjarsdóttir. Mapping the landscape of sustainable hci. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1975–1984, 2010.

[24] Elad Elrom. *Integrating D3. js with React*. Springer, 2021.

[25] Raymond F Enriquez. *New Basics of Computer Graphics 2020*. Creative Hands Publishing, 2020.

[26] Jennifer Fereday and Eimear Muir-Cochrane. Demonstrating rigor using thematic analysis: A hybrid approach of inductive and deductive coding and theme development. *International Journal of Qualitative Methods*, 5(1):80–92, March 2006.

[27] Laura Fortunato and Mark Galassi. The case for free and open source software in research and scholarship. *Philosophical Transactions of the Royal Society A*, 379(2197):20200079, 2021.

[28] Lon Åke Erni Johannes Hansson, Teresa Cerratto Pargman, and Daniel Sapiens Pargman. A decade of sustainable hci: connecting shci

to the sustainable development goals. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–19, 2021.

[29] Nikolay Harutyunyan, Andreas Bauer, and Dirk Riehle. Industry requirements for floss governance tools to facilitate the use of open source software in commercial products. *Journal of Systems and Software*, 158:110390, 2019.

[30] Julian Holfeld. On the relevance of the godot engine in the indie game development industry, 2024.

[31] Afzal Hussain, Haad Shakeel, Faizan Hussain, Nasir Uddin, and Turab Latif Ghouri. Unity game development engine: A technical survey. *Univ. Sindh J. Inf. Commun. Technol*, 4(2):73–81, 2020.

[32] Björn A Johnsson, Martin Höst, and Boris Magnusson. Evaluating a gui development tool for internet of things and android. In *International Conference on Product-Focused Software Process Improvement*, pages 181–197. Springer, 2016.

[33] Linnea Kirby, Paul Buser, and Marcelo M. Wanderley. Introducing the t-tree: Using multiple t-sticks for performance and installation. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, The University of Auckland, New Zealand, June 2022.

[34] Dmitry Kirsanov. *The book of inkscape 2nd edition*. No Starch Press, San Francisco, CA, December 2021.

[35] Andrzej Klimczuk, Vida Česnuityte, and Gabriela Avram. *The Collaborative Economy in Action: European Perspectives*. University of Limerick, October 2021.

[36] Susan Lechelt, Katerina Gorkovenko, Luis Lourenço Soares, Chris Speed, James K. Thorp, and Michael Stead. Designing for the end of life of iot objects. In *Companion Publication of the 2020 ACM Designing Interactive Systems Conference*, DIS' 20 Companion, page 417–420, New York, NY, USA, 2020. Association for Computing Machinery.

[37] Jennifer Mankoff. Hci and sustainability: A tale of two motivations. *Interactions*, 19(3):16–19, may 2012.

[38] Raul Masu, Mela Bettega, Nuno N Correia, and Teresa Romão. Investigating performance ecologies using screen scores: a case study. *Personal and Ubiquitous Computing*, 27(5):1887–1907, 2023.

[39] Raul Masu, Nuno N Correia, and Teresa Romão. Technology-mediated musical connections: the ecology of a screen-score performance. In *Proceedings of the 16th International Audio Mostly Conference*, pages 109–116, 2021.

[40] Raul Masu, Adam Pultz Melbye, John Sullivan, and Alexander Refsum Jensenius. Nime and the environment: toward a more sustainable nime practice. In *NIME 2021*. PubPub, 2021.

[41] Raul Masu, Nicolò Merendino, Antonio Rodà, and Luca Turchet. Sustainable internet of musical things: Strategies to account for environmental and social sustainability in network-based interactive music systems. *IEEE Access*, 2024.

[42] Nicolo Merendino, Mela Bettega, Adam Pultz Melbye, John Sullivan, Antonio Rodà, and Raul Masu. Sustainable digital fabrication in nime: Nine sustainability strategies for dmi production. In *NIME 2024*, 2024.

[43] Nicolo' Merendino, Lepri Giacomo, Antonio Rodà, Masu Raul, et al. Redesigning the chowndolo: a reflection-on-action analysis to identify sustainable strategies for nimes design. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2023.

[44] Nicolò Merendino and Antonio Rodà. Defining an open source cad workflow for experimental music and media arts. In *10th International Conference on Digital and Interactive Arts*, pages 1–6, 2021.

[45] Nicoló Merendino, Antonio Rodà, and Raul Masu. "below 58 bpm," involving real-time monitoring and self-medication practices in music performance through iot technology. *Frontiers in Computer Science*, 6:1187933, 2024.

[46] Pultz Melbe Sullivan Rodà Masu Merendino, Bettega. Sustainable digital fabrication in nime: Nine sustainability strategies for dmi production. In *Proceedings of NIME 2024*, 2024.

[47] Fabio Morreale and Andrew McPherson. Design for longevity: Ongoing use of instruments from nime 2010-14. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 192–197, Copenhagen, Denmark, 2017. Aalborg University Copenhagen.

[48] Manoj Muniswamaiah, Tilak Agerwala, and Charles C. Tappert. Green computing for internet of things. In *2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pages 182–185, 2020.

[49] Waqas Nawaz and Muammer Koç. Development of a systematic framework for sustainability management of organizations. *Journal of cleaner production*, 171:1255–1274, 2018.

[50] Alex Nieva, Johnty Wang, Joseph Malloch, and Marcelo Wanderley. The t-stick: Maintaining a 12 year-old digital musical instrument. In Thomas Martin Luke Dahl, Douglas Bowman, editor, *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 198–199, Blacksburg, Virginia, USA, June 2018. Virginia Tech.

[51] Joshua M. Pearce. Economic savings for scientific free and open source technology: A review. *HardwareX*, 8:e00139, 2020.

[52] Lasse Apalnes Pedersen, Svavar Skuli Einarsson, Fredrik Arne Rikheim, and Frode Eika Sandnes. *User Interfaces in Dark Mode During Daytime – Improved Productivity or Just Cool-Looking?*, page 178–187. Springer International Publishing, 2020.

[53] Ondřej Pešička. Dark mode ui design – 7 best practices, 2023.

[54] Daphne Plaizier. Creating warmth through visual hierarchy in ux design; enhancing engagement and social connectedness with a charity website, February 2024.

[55] Kevin Proudfoot. Inductive/deductive hybrid thematic analysis in mixed methods research. *Journal of Mixed Methods Research*, 17(3):308–326, September 2022.

[56] Nicholas P Radtke, Marco A Janssen, and James S Collofello. What makes free/libre open source software (floss) projects successful? an agent-based model of floss projects. *International Journal of Open Source Software and Processes (IJOSSP)*, 1(2):1–13, 2009.

[57] Roel Roscam Abbing. On cultivating the installable base. In *Proceedings of the Participatory Design Conference 2022-Volume 2*, pages 203–207, 2022.

[58] Roel Roscam Abbing and Ann Light. Make friends not art: Mapping law, power and participation in designing an online platform during documenta fifteen. In *Proceedings of the Participatory Design Conference 2024: Full Papers - Volume 1*, PDC '24, page 86–97, New York, NY, USA, 2024. Association for Computing Machinery.

[59] Adnan Sabovic, Ashish Kumar Sultania, and Jeroen Famaey. Demonstration of an energy-aware task scheduler for battery-less iot devices. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*, SenSys '21, page 586–587, New York, NY, USA, 2021. Association for Computing Machinery.

[60] Andrew Sanders. *An introduction to Unreal engine 4*. AK Peters/CRC Press, 2016.

[61] Jenna Scaglione. Why do people use dark mode?, 2022.

[62] Mahyar Shirvanimoghaddam, Kamyar Shirvanimoghaddam, Mohammad Mahdi Abolhasani, Majid Farhangi, Vahid Zahiri Barsari, Hangyue Liu, Mischa Dohler, and Minoo Naebe. Towards a green and self-powered internet of things using piezoelectric energy harvesting. *IEEE Access*, 7:94533–94556, 2019.

[63] Fernanda Gomes Silva, Paulo Ezequiel Dias dos Santos, and Christina von Flach G. Chavez. Do we use floss in software engineering education? mapping the profiles and practices of higher education teachers from brazil. In *Proceedings of the XXXIV Brazilian Symposium on Software Engineering*, pages 473–482, 2020.

[64] Krista Sorri, Navonil Mustafee, and Marko Seppänen. Revisiting iot definitions: A framework towards comprehensive use. *Technological Forecasting and Social Change*, 179:121623, 2022.

[65] Krista Sorri, Navonil Mustafee, and Marko Seppänen. Revisiting iot definitions: A framework towards comprehensive use. *Technological Forecasting and Social Change*, 179:121623, 2022.

[66] Sruthi. Fonts in ux design, 2022.

[67] Weronika Stachurska, Aleksandra Witoszek-Kubicka, and Magdalena Igras-Cybulska. Virtual reality in music education: A qualitative user study of harmospherevr. In *Proceedings of the Workshop on Prototyping and Developing Real-World Applications of Extended Reality at the 17th International Conference on Advanced Visual Interfaces (Genoa, Italy)*, 2024.

[68] Pannee Suanpang, Chawalin Niamsorn, Pattanaphong Pothipassa, Thinnagorn Chunhapataragul, Titiya Netwong, and Kittisak Jermsittiparsert. Extensible metaverse implication for a smart tourism city. *Sustainability*, 14(21), 2022.

[69] Ashish Kumar Sultania and Jeroen Famaey. Energy-aware battery-less bluetooth low energy device prototype powered by ambient light. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*, SenSys '21, page 584–585, New York, NY, USA, 2021. Association for Computing Machinery.

[70] David Trenholme and Shamus P Smith. Computer game engines for developing first-person virtual environments. *Virtual reality*, 12:181–187, 2008.
[71] Luca Turchet and Francesco Antoniazzi. Semantic web of musical things: Achieving interoperability in the internet of musical things. *Journal of Web Semantics*, 75:100758, 2023.
[72] Luca Turchet, Carlo Fischione, Georg Essl, Damián Keller, and Mathieu Barthet. Internet of musical things: Vision and challenges. *Ieee access*, 6:61994–62017, 2018.
[73] Melissa Wen, Leonardo Leite, Fabio Kon, and Paulo Meirelles. Understanding floss through community publications: strategies for grey literature review. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: New Ideas and Emerging Results*, pages 89–92, 2020.
[74] David A Wheeler. Why free-libre/open source software (floss)? look at the numbers! *Citeseer*, 2011.
[75] Azeema Yaseen, Sutirtha Chakraborty, and Joseph Timoney. A cooperative and interactive gesture-based drumming interface with application to the internet of musical things. In *International Conference on Human-Computer Interaction*, pages 85–92. Springer, 2022.
[76] Hao Zhang. Iot product design for user experience and technological innovation in virtual reality environments. *EAI Endorsed Transactions on Scalable Information Systems*, 11(5), 2024.
[77] Yan Zhang. The effect of font design characteristics on font legibility. Master's thesis, Concordia University, 2006. Unpublished.