Delft University of Technology

Sustainable Software Engineering
CS4415

Susie: A Tool for Evaluating GitHub Repository Sustainability

Authors Group 9:
Philippe de Bekker (4876385)
Merel Steenbergen (4784871)
Sebastien van Tiggele (4705165)
Ivor Zagorac (4691202)

March 31, 2023

Contents

1 Introduction

The ICT sector is projected to account for 14% of the global carbon footprint by 2040 [1]. It is the responsibility of software developers to start thinking about sustainability in their projects. The impact on the carbon footprint mostly has to do with environmental sustainability, but there are four more forms of sustainability that are just as important for resilience. The other four perspectives are: social, individual, economic, and technical. Social and individual sustainability focus on people. While social sustainability is more concerned with society and organisations, individual sustainability focuses on the well-being of the people in an organisation and how they interact. Economic sustainability means generating profit such that the project can keep existing. And finally, technical sustainability means that the project is well-maintained.

Existing research on sustainable software engineering has mainly focused on developing guidelines and best practices for building sustainable software from scratch, but there is little information on how to assess the sustainability of already existing projects. By providing an analysis of the sustainability of GitHub repositories, our website can help developers and other stakeholders make more informed decisions about the sustainability of the software they use and contribute to. Additionally, our website can help raise awareness about sustainable software engineering practices and encourage developers to adopt more sustainable habits in their future projects. Even though the projects might not be changed anymore, maybe the developing habits still can.

section 2 will provide the details of our proposal. The implementation will be discussed in section 3, after which the validation and social impact will be explained in sections 4 and 5. Finally, a brief overview and recap will be given in section 6.

2 Proposal

We would like to introduce Susie: A website designed to analyze the social, individual, technical and environmental sustainability, of public GitHub repositories. It allows users to provide the link to their GitHub repository and receive a detailed analysis of its sustainability. This sustainability is measured according to several metrics, which will be explained in subsection 2.1. It also includes guides for helping developers make more sustainable choices for their next projects.

2.1 Metrics

Upon entering the link to a public repository, Susie uses various metrics to generate a sustainability report. This report includes information on the energy consumption of the repository's code, an assessment of the social sustainability of the development practices used in the project and an analysis of the individual sustainability of the developers working on the project.

2.1.1 Programming Languages

Programming languages can vary significantly in terms of energy consumption. As such, understanding the energy consumption of programming languages is crucial to reduce the carbon footprint and promoting sustainable practices in software development. [5] shows a table with an energy-consumption score for several programming languages. Susie uses these scores to give advice on the programming language used in a repository. For example, plain JavaScript is about five times more efficient than TypeScript, so repositories with many lines of TypeScript will receive advice to consider JavaScript for future projects.

Programming Language	Energy Consumption (MJ/LOC)
C	1
Rust	1.03
C++	1.34
Ada	1.70
Java	1.98
Pascal	2.14
Chapel	2.18
Lisp	2.27
Ocaml	2.40
Fortran	2.52
Swift	2.79
Haskell	3.10
C#	3.14
Go	3.23
Dart	3.83
F#	4.13
JavaScript	4.45
Racket	7.91
TypeScript	21.50
Hack	24.02
PHP	29.30
Erlang	42.23
Lua	45.98
Jruby	46.54
Ruby	69.91
Python	75.88
Perl	79.58

Table 1: Energy Consumption of Programming Languages

2.1.2 Inclusive language

Inclusive language is crucial in GitHub repositories because it helps to create a welcoming and respectful environment for all contributors, regardless of their background or identity. Using inclusive language means choosing words and phrases that do not exclude or marginalize certain groups of people. This includes avoiding the use of gendered pronouns or terms that may be offensive or insensitive to particular cultures or identities. By using inclusive language in GitHub repositories, developers can foster a more diverse and inclusive community that encourages participation from individuals of all backgrounds, ultimately leading to better collaboration, innovation, and success of the project. Additionally, using inclusive language can also help to mitigate the risk of misunderstandings or conflicts that may arise from insensitive language use. In a blog post by the Academy Software Foundation, the organisation explains how to use inclusive language in programming [3].

Susie checks your repository for common terms that may be considered as not inclusive. Table 2.1.2 shows part of the list which is used for inclusive language checks in the analysed GitHub repository.[1] It provides examples of common terms and phrases that may be considered insensitive or exclusionary to certain groups of people.

[1]For the full list, check the implementation on `https://github.com/philippedeb/susie`

Common Terms/Phrases	Inclusive Suggestions
master/slave	main/replica, leader/follower, primary/secondary
whitelist	allow list, inclusion list, safe list
blacklist	deny list, exclusion list, block list, banned list
man hours	labor hours, work hours, person hours, engineer hours
manpower	labor, workforce
guys	folks, people, you all
girl/girls	woman/women
middleman	middle person, mediator, liaison
he/she, him/her, his/hers	they, them, theirs
crazy/insane	unpredictable, unexpected
normal/abnormal	typical/atypical
grandfather/grandfathering/legacy	flagship, established, rollover, carryover
crushing it/killing it	elevating, exceeding expectations, excelling
owner	lead, manager, expert
sanity check	quick check, confidence check, coherence
dummy value	placeholder value, sample value
native feature	core feature, built-in feature
culture fit	values fit
housekeeping	cleanup, maintenance

Table 2: List of suggestions for inclusive language

2.1.3 Governance

The 'Governance' metric aims to check several aspects related to the governance and sustainability of an open-source project. This metric examines the presence of a `README.md` file, license, changelog, code of conduct, contributing guidelines, issue template, and pull request template. Each item is checked for completeness, with a green tick indicating that the item is present and a red cross indicating that it is not. For a visual example, please refer to Figure 8.

One additional check that the 'Governance' metric performs is to search for sustainability-related keywords in the `README.md` file. The terms that Susie (currently) takes into account for sustainability are as follows:

- biodegradable
- carbon emission
- carbon footprint
- carbon neutral
- carbon offset
- carbon positive
- circular economy
- climate action
- climate change
- ecological footprint
- ecological impact
- e-waste
- energy consumption
- energy efficiency
- energy saving
- energy statement
- environmental impact
- landfill-free
- leed certification
- organic
- paris agreement
- pollution
- recyclable
- recycling
- sustainability
- sustainable
- waste-to-energy
- waste-to-profit
- zero carbon
- zero waste

If any of these terms are found, the metric indicates that the project addresses sustainability concerns. The aforementioned terms are based on the most popular search terms and glossaries of sustainability terms and definitions [4, 7, 2]. Currently, the terms are handpicked based on intuition, however, for optimal effectiveness, the usage of each individual term should be reviewed by domain experts.

Overall, the Governance metric helps users evaluate the governance practices of open-source projects and their commitment to sustainability. By providing a clear and concise overview of these aspects, Susie aims to help users make more informed decisions about which projects to contribute to or use in their own work.

2.1.4 Workflow analysis

Running workflows locally before executing them remotely is beneficial for saving on energy usage. Executing workflows remotely involves utilizing resources such as servers and data centres that require a significant amount of energy compared to running them locally. Therefore, Susie analyses how many workflows have passed and failed in the last hundred runs. If more than ten percent of the builds failed, the user will receive a warning that this is quite significant and that the best practice is to always run and fix builds locally before pushing.

2.1.5 Issue Sentiment Analysis

Sentiment analysis can be applied to the comments and interactions within the issues in GitHub repositories to gain insights into the emotional tone of discussions between developers working on a project. Issues are used to report problems or suggest new features in software development projects, and analyzing the sentiment of these interactions can provide valuable insights into the overall health and sustainability of the project. By analyzing the language used in issue comments and interactions, sentiment analysis can help identify patterns of positive or negative sentiment among developers working on the project. For example, if there are a large number of negative comments on an issue, this may indicate that the issue is causing frustration among the development team and may need to be addressed quickly to avoid potential problems.

Susie uses the npm package 'sentiment' to perform sentiment analysis on text data from issues in Github repositories [6]. The sentiment package analyzes the text and assigns sentiment scores to individual words, and then uses these scores to determine the overall sentiment of the text. Susie uses this information to provide insights on the sentiment of issues in GitHub repositories.

2.2 Guides

Susie's 'Guides' page is a valuable resource for developers because it provides them with actionable steps and practical advice on how to make their software more sustainable. The guides cover a range of topics related to sustainability in software development, including energy efficiency and inclusive language for social sustainability, and are designed to be accessible and easy to understand.

One of the guides, the energy efficiency guide, specifically focuses on the use of programming languages to optimize code for energy efficiency. This guide highlights the importance of selecting programming languages that are optimized for energy efficiency to reduce the carbon footprint and promote sustainable practices in software development. The guide includes an informative table that provides an overview of the energy consumption scores for different programming languages, allowing developers to make informed decisions when selecting programming languages for their projects. By providing this information, Susie is helping to raise awareness about the

energy impact of software development and empowering developers to make more sustainable choices in their work. The inclusive language for social sustainability guide, on the other hand, provides guidance on how to use language in a way that is respectful, inclusive, and avoids perpetuating harmful stereotypes or biases.

Currently, the guides only cover these topics. However, because the Susie project will be open source, other developers are encouraged to contribute to the project and add more guides to the page. This open-source approach allows for a collaborative effort towards sustainability education and helps to ensure that the Susie 'Guides' page remains relevant and up-to-date with the latest sustainability practices in software engineering. By encouraging contributions from a diverse range of developers, Susie is fostering a community of sustainable software engineers who are dedicated to creating a more sustainable future.

3 Implementation

Susie has been implemented using a modern tech stack that includes TypeScript, React, Bootstrap, and Vite, as well as several additional npm packages that provide functionality such as data visualization. The use of modern web technologies like React and Bootstrap, albeit not the most sustainable solution (as mentioned in Section 2.1.1), ensures that the site is responsive and user-friendly on a wide range of devices and screen sizes. Furthermore, the React components have been developed in such a way that adding more dashboard sections or new guides is automatically recognized by other components and scales the website effortlessly, making new releases immensely feasible. To reach users all around the world, Susie has been deployed on GitHub Pages at `https://philippedeb.github.io/susie/`.

The website is organized into several main sections, each of which serves a distinct purpose. The landing page of Susie allows users to browse different sections of the website and prompts users to enter the link to a GitHub repository that they want to analyse, see Figure 1.

Once the analysis is complete, the results are displayed in the dashboard section, which provides detailed information about the sustainability of the repository according to several key metrics (see Section 2.1). Finally, the guides section provides resources and best practices for developers who want to create more sustainable software in the future, see Section 2.2 and Figure 2 below.

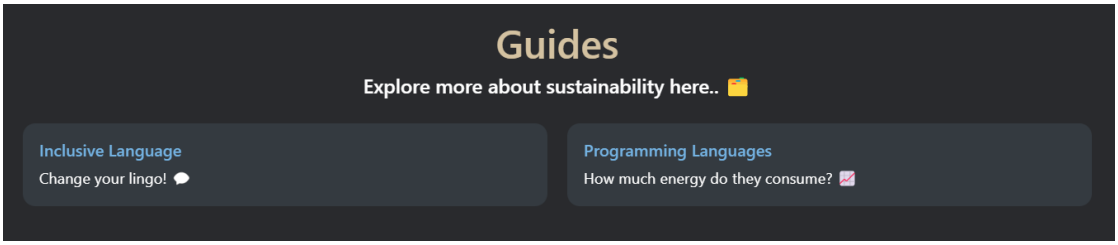

Figure 2: Guides section of Susie - to learn more about sustainable software development.

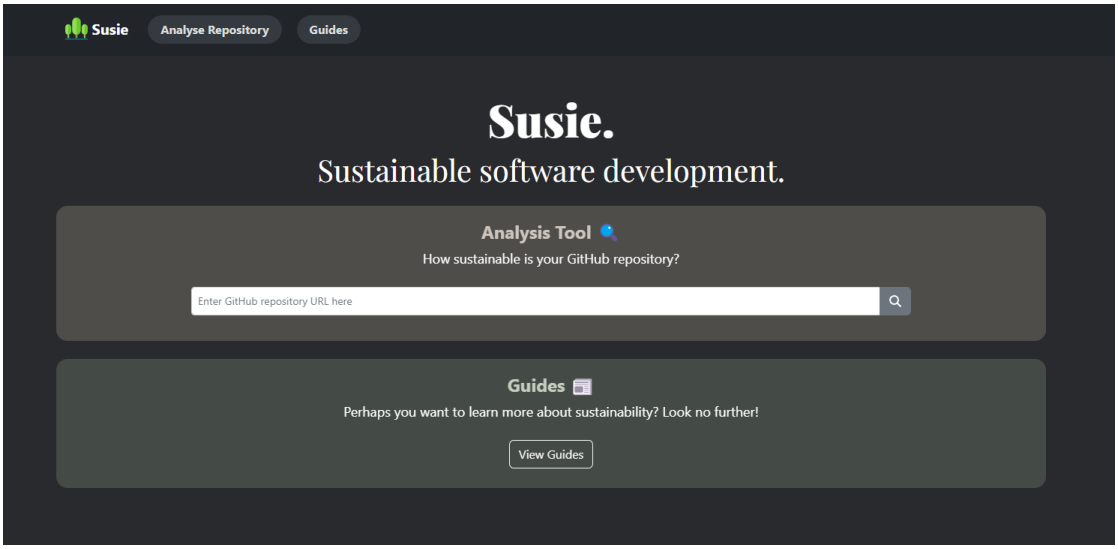

Figure 1: Landing page of Susie. Users can browse different sections and are prompted to enter a link to a GitHub repository to analyse.

Overall, the Susie website has a clean and intuitive layout that makes it easy for users to navigate and understand. Moreover, to save energy, Susie uses a dark theme throughout the website. For a complete visual overview of Susie, please refer to Appendix A or check out the actual website online.

4 Validation

We performed a validation study on a sample of GitHub repositories. The validation study aimed to assess Susie's ability to accurately identify repositories that incorporate sustainable software engineering practices. Concrete data regarding the validation can be found in Appendix B.

In general, Susie's effectiveness in assessing the sustainability of software engineering practices was already demonstrated in Appendix A and works exactly as described for most metrics in subsection 2.1. However, there are a few fascinating cases to depict regarding the validation of Susie that would otherwise perhaps go unnoticed. On the repository *Zero Bullshit Haskell*[2], Susie was successfully able to detect profane language and other non-inclusive language usages, as shown in Figure 12. Furthermore, we also analyzed the repository *GPT4ALL*[3], which was correctly not deemed as sustainable. However, Susie should become more aware of big machine learning and artificial intelligence repositories to properly score the sustainability on the impact of such repositories, and not just on keywords in the `README.md` file. Our sentiment analysis in this repository revealed a high rate of false positives due to software engineering jargon using terms like 'support' and 'illegal' in a different way than normal contexts, as illustrated in Figure 13 and Figure 14.

Overall, the results of the validation study partly confirm the utility of Susie as a powerful tool for developers and other stakeholders to assess the sustainability of their software engineering

[2] `https://github.com/alpacaaa/zero-bullshit-haskell`
[3] `https://github.com/nomic-ai/gpt4all`

practices, however, has drawbacks in the area of sentiment analysis and checking if a repository concerns sustainability.

5 Social impact

To achieve our goal of promoting sustainable software engineering practices and increasing awareness among developers, it is important that we take measures to maximize the social impact of our tool. One of the key ways in which we plan to accomplish this is by making Susie an open-source project that enables easy and meaningful contributions from any developer. By providing a transparent and collaborative platform for developers to work together and contribute towards sustainable software engineering practices, we hope to create a culture of sustainability that extends beyond individual software projects. Additionally, we recognize the importance of using social media as a tool for raising awareness and promoting sustainable software engineering practices to a wider audience.

6 Conclusion

All things considered, Susie is a powerful tool for analyzing the sustainability of public GitHub repositories, with a focus on social, individual, technical, economic, and environmental sustainability. By using a wide range of metrics (i.e. programming languages, inclusive language, governance, workflows, issue sentiment), Susie generates detailed audits that enable developers and other stakeholders to make informed decisions about the sustainability of the software they use and/or contribute to. With also the addition of guides on the website, Susie promotes sustainable software engineering practices and encourages developers to adopt more sustainable habits in their future projects. To fully incorporate the sustainability movement in the community, Susie has become an open-source project that welcomes contributions from any developer, such that we can create and speed up this journey together. Make sure to join at `https://github.com/philippedeb/susie`!

7 Future work

7.0.1 GitHub token

When making requests to the GitHub REST API without a token, they are considered anonymous requests, meaning that GitHub treats the request as if it is coming from an unauthenticated user. This means that users will quickly run into the rate-limit that GitHub enforces, which can limit the number of requests that can be made per hour. These anonymous requests have been used in this project, as a token costs money that was not available for this project.

Therefore, after a few tries of running Susie on a repository, the rate-limit for unauthorized requests is reached. For this prototype version, it is acceptable, however, if the project would be used on a larger scale, a token would be a necessity.

7.0.2 Dockerfile analysis

Many projects are deployed using Dockerfiles. These can be analysed for sustainability. Several aspects of the file can impact this. First, the image size plays a large role in the amount of storage space required and the time it takes to transfer it. Data storage and transmission can cost quite

a lot of energy, so by optimizing the size of the image and reducing unnecessary dependencies, we can reduce the environmental impact of a Dockerfile.

Furthermore, Docker images may consume many resources such as CPU, memory, and disk space, which can lead to higher energy consumption and longer build times. By optimizing resource utilization, we can reduce the environmental impact of the Dockerfile.

Moreover, image build time also has an impact on the environment. By reducing the build time through techniques such as caching and layering, we can reduce the environmental impact of the Dockerfile.

This feature was not implemented yet, because all of the project team members did not have significant experience with Docker yet. We therefore decided to focus on other features and metrics. However, it would be a desirable feature to have for Susie and is thus still an open issue in the GitHub repository.

7.0.3 Geolocation analysis

Using geolocation data from GitHub comments can provide an indication of the contributor diversity working on a repository. By analysing this data we could gain insight into the diversity of the contributors. A high diversity of contributors can indicate that the project has got people working on it of different backgrounds, having different experiences and perspectives that can lead to a more inclusive product. This would also indicate that the awareness of a repository is more spread out.

Relying solely on the geolocation data of GitHub commits might be an unreliable way of determining contributor diversity, and more research is needed to incorporate it in the GitHub data analysis. However, with the right methods and motivations it could be a good metric to show on Susie, and therefore it is something future developers should definitely consider.

References

[1] Luis Cruz. *Sustainable Software Engineering*. https://luiscruz.github.io/course_sustainableSE/2022/. Accessed: 29-03-2023.

[2] *Environmental Keywords - Find SEO Google AdWords Key Words for Your Website*. Dec. 29, 2021. URL: https://www.wordstream.com/popular-keywords/environmental-keywords (visited on 03/29/2023).

[3] Academy Software Foundation. *Inclusive language in technology*. Feb. 2021. URL: https://www.aswf.io/blog/inclusive-language/.

[4] *Glossary of sustainability*. URL: https://sustainable.org.nz/learn/tools-resources/glossary-of-sustainability/ (visited on 03/29/2023).

[5] Rui Pereira et al. "Energy Efficiency across Programming Languages: How Do Energy, Time, and Memory Relate?" In: *Proceedings of the 10th ACM SIGPLAN International Conference on Software Language Engineering*. SLE 2017. Vancouver, BC, Canada: Association for Computing Machinery, 2017, pp. 256–267. ISBN: 9781450355254. DOI: 10.1145/3136014.3136031. URL: https://doi.org/10.1145/3136014.3136031.

[6] *Sentiment*. URL: https://www.npmjs.com/package/sentiment.

[7] Sustainable Review. *Sustainability 101: 35 terms and definitions you need to know*. Feb. 6, 2023. URL: https://sustainablereview.com/sustainability-101-terms-and-definitions/ (visited on 03/29/2023).

A Visual overview of Susie

This appendix provides a visual overview of the website. The figures included in this section show the various parts of the website, such that readers can understand the layout and functionality of the tool more easily and can provide a more concrete sense of how Susie works in practice. For the landing page and guides section, please refer to Figure 1 and Figure 2, respectively. As an example, a popular Python library named `Streamlit`[4] is used to demonstrate Susie below. Please read the captions in order to have an optimal storyline.

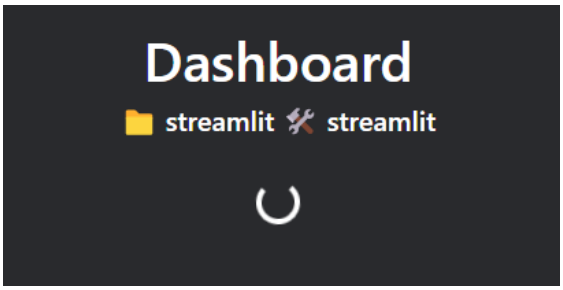

Figure 3: When Susie is analysing a repository, an animated loading icon is displayed.

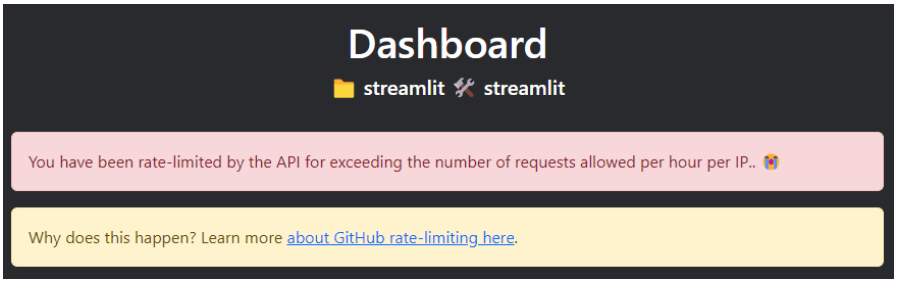

Figure 4: When Susie fails to analyse a repository due to various reasons, the user gets an alert with the reason why Susie failed. In this example, the user has analysed too many repositories in a short timeframe and the GitHub API has rate-limited the user, making Susie temporarily unusable.

[4] `https://www.github.com/streamlit/streamlit`

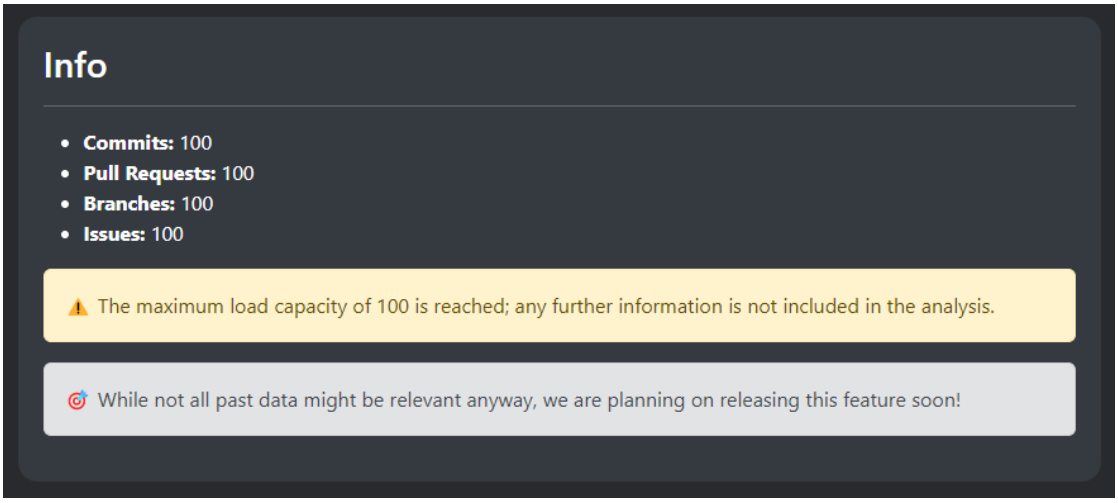

Figure 5: When the repository is analysed, Susie displays several sections in a dashboard. The first one is *Info*, which gives a brief overview of the repository. Additionally, it shows any warnings regarding fetching the data.

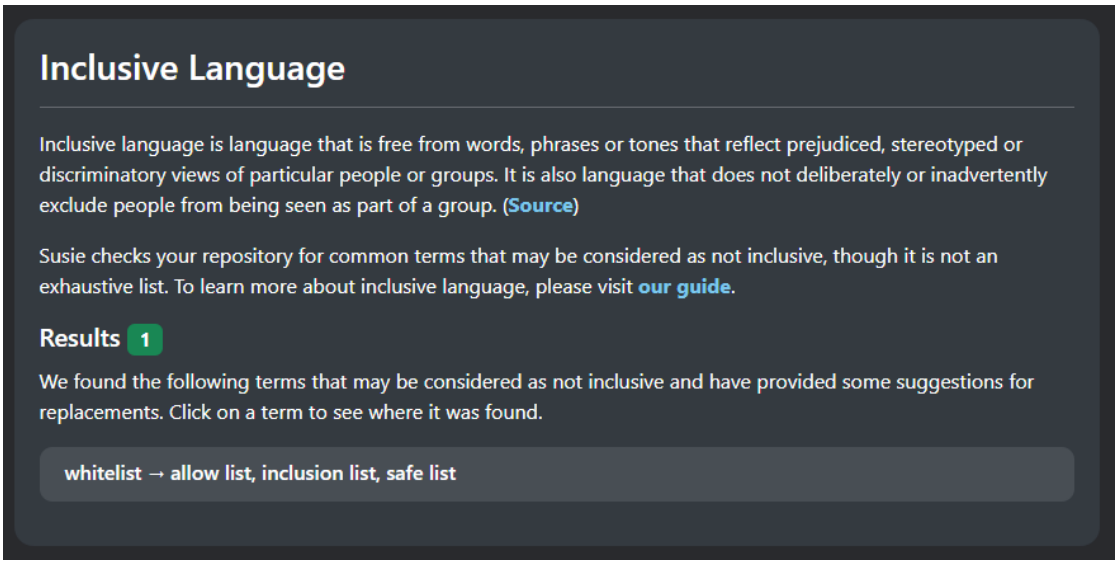

Figure 6: The *Inclusive Language* section in the dashboard. Any suggestion can be expanded by the means of a dropdown component and will list the origins of the respective suggestion.

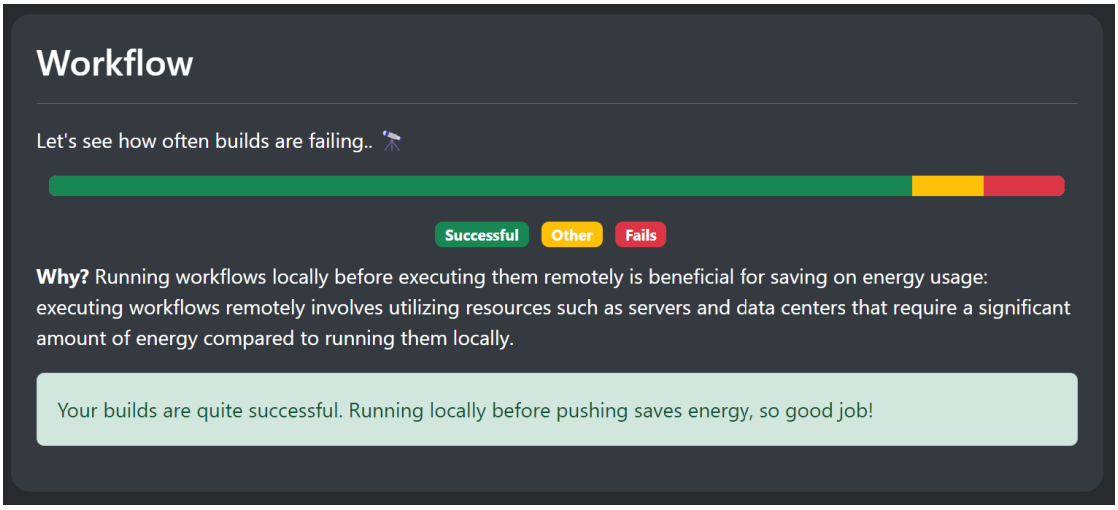

Figure 7: The *Workflow* section in the dashboard. If no data is found, the bar is grey and has no legend.

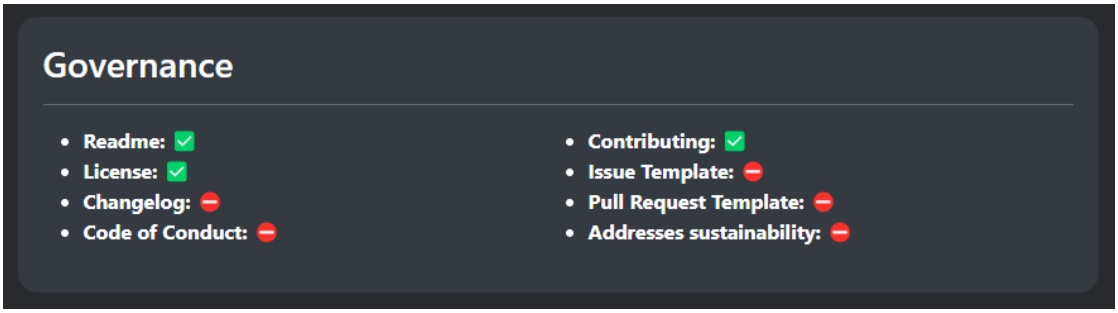

Figure 8: The *Governance* section in the dashboard.

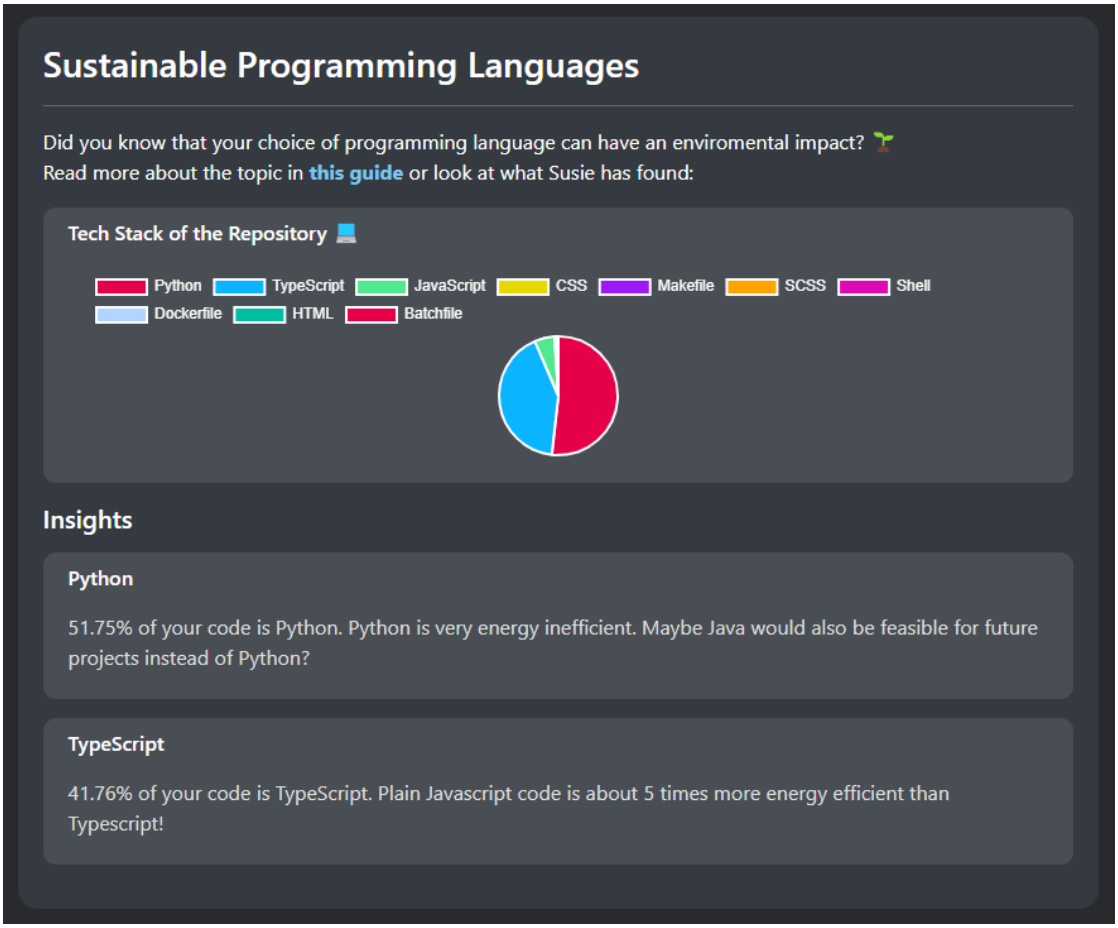

Figure 9: The *Sustainable Programming Languages* section in the dashboard. An animated and responsive piechart using `Chart.js` shows the tech stack and Susie provides insights for the most prominent languages used in the GitHub repository. All inner sections can be minimized by clicking on the bold titles, in case the user finds the number of details becoming too chaotic when browsing through the Susie dashboard.

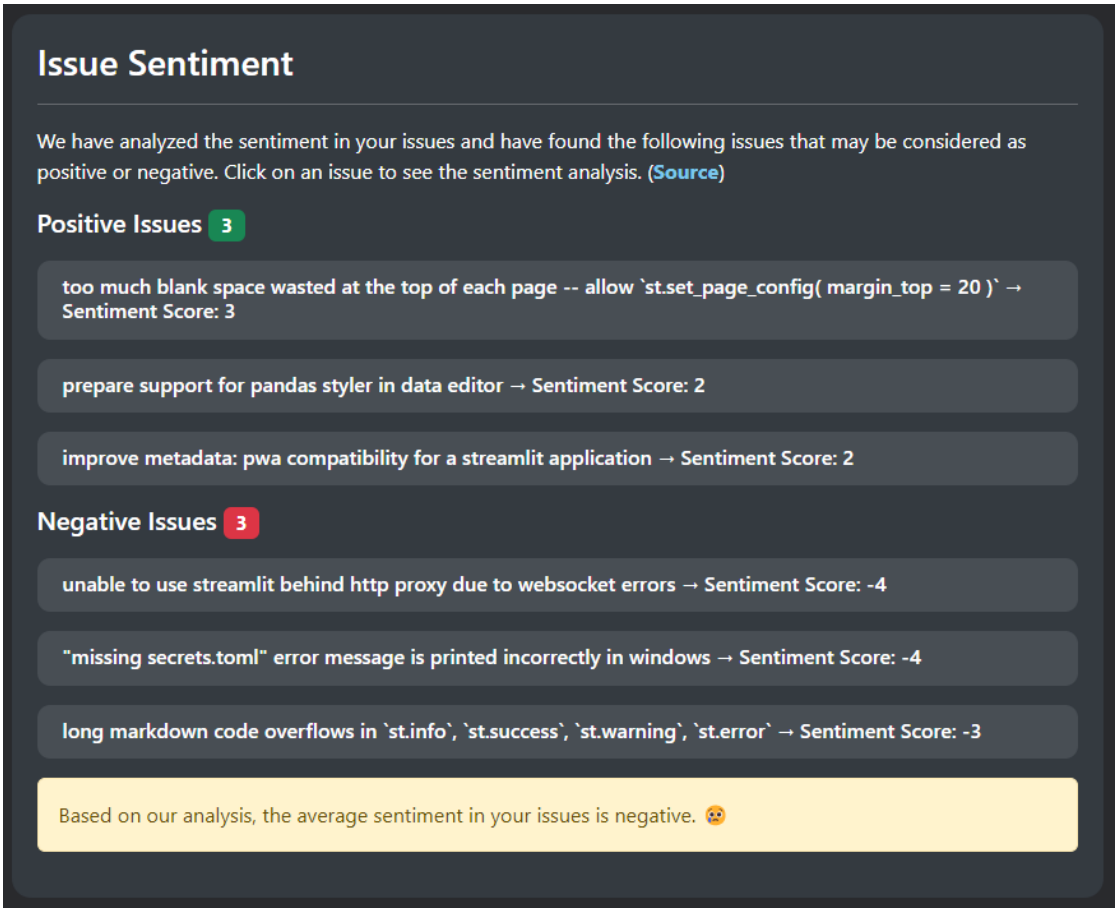

Figure 10: The *Issue Sentiment* section in the dashboard.

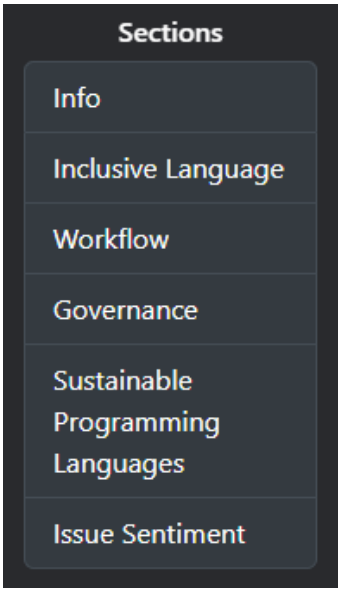

Figure 11: With so many sections in the dashboard, Susie also provides a floating sidebar on non-mobile devices to smoothly browse through the dashboard (animated scroll).

B Validation Samples

This appendix provides additional validation samples for the metrics used in Susie. Please refer to section 4 for more information.

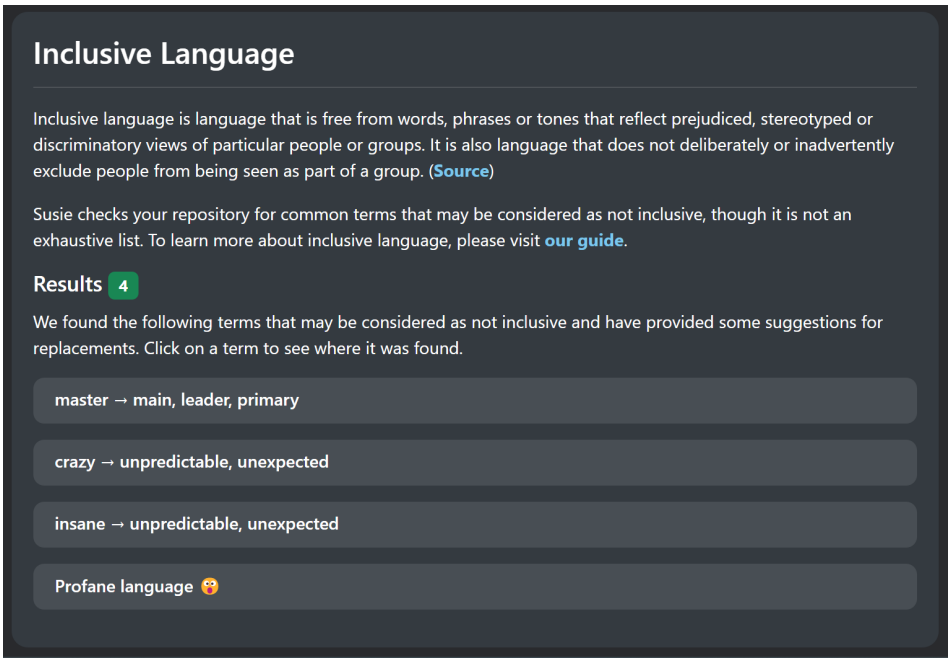

Figure 12: Validation sample for the inclusive language metric.

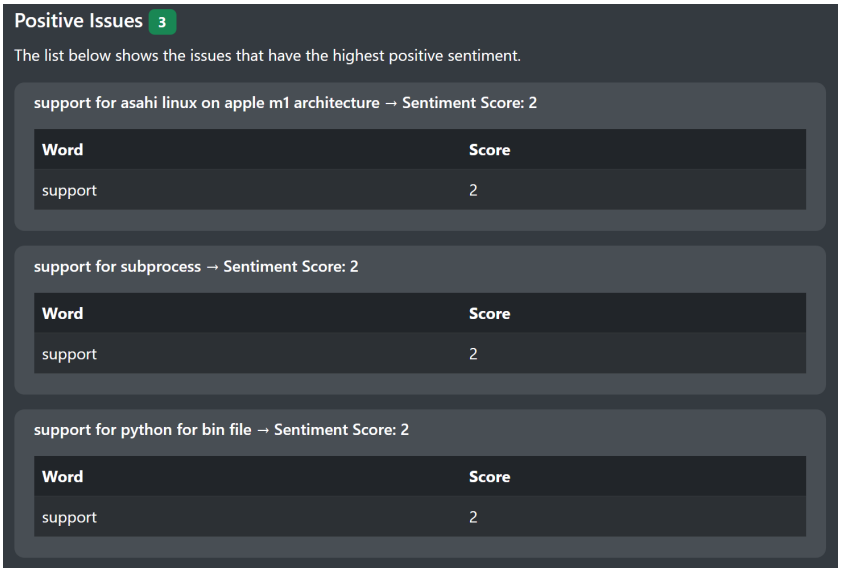

Figure 13: Validation sample for the issue sentiment metric (in this case, positive sentiment).

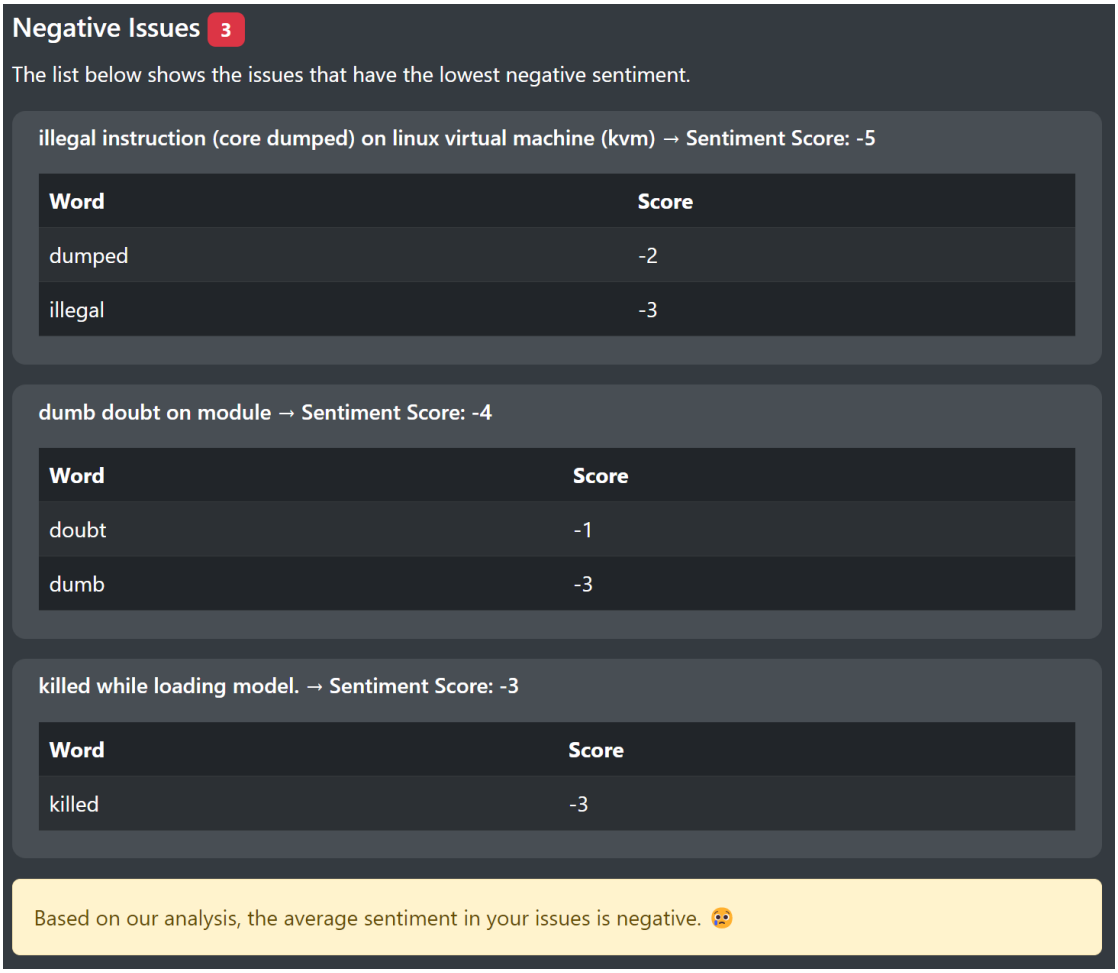

Figure 14: Validation sample for the issue sentiment metric (in this case, negative sentiment).