A Catalog of Green Software Measures Mapped to the Scrum Framework

ID	Scrum phase	Measure	GSMM Axes	Maturity Level	Impact Level	Sources
P-1	Sprint Planning	Select cloud regions powered by greener energy.	Energy, Commitments	2	4	[1]
P-2	Sprint Planning	Choose hosting providers with credible renewable energy sourcing.	Energy, Commitments	2	4	[1], [2], [3]
P-3	Sprint Planning	Choose a software platform that is developing in a sustainable direction.	Products	2	3	[1]
P-4	Sprint Planning	Consider utilizing open-source projects instead of proprietary software.	Products, Training	2	3	[1]
P-5	Sprint Planning	Consider the environmental tradeoffs that come with monolithic and distributed architectures.	Products, Carbon Ops	2	3	[1]
P-6	Sprint Planning	Critically evaluate the necessity of third-party services.	Products, Carbon Ops	2	3	[2], [4]
P-7	Sprint Planning	Critically assess the necessity of specialized hardware.	Devices, Utilization	2	3	[1]
P-8	Sprint Planning	Consider using existing open-source datasets for ML.	Products, Training	2	3	[1]
P-9	Sprint Planning	Participate in a platform's community that is pushing for sustainability.	Training, Commitments	2	2	[1]
P-10	Sprint Planning	Implement caching strategies at the browser, server or CDN level to reduce redundant API calls.	Carbon Ops, Energy, Utilization	3	5	[1], [3], [4]
P-11	Sprint Planning	Consider switching to cloud hosting providers and using multitenancy options.	Utilization, Energy	3	5	[1], [3]
P-12	Sprint Planning	When choosing programming languages, consider their energy profiles.	Products, Carbon Ops	3	4	[1], [3], [4]
P-13	Sprint Planning	When choosing to use frameworks and libraries, consider how they affect the sustainability of the project.	Products, Carbon Ops	3	4	[1], [2], [4]
P-14	Sprint Planning	Avoid excessive layering when making architectural decision.	Carbon Ops	3	4	[1]
P-15	Sprint Planning	Reduce synchronous calls between services.	Carbon Ops, Energy	3	4	[1]

ID	Scrum phase	Measure	GSMM Axes	Maturity Level	Impact Level	Sources
P-16	Sprint Planning	Leverage Progressive Web Apps (PWAs) with caching strategies and service workers.	Carbon Ops, Energy, Devices	3	4	[3], [4]
P-17	Sprint Planning	When choosing any of the web services the applications will depend on, apply the principles of green hosting.	Products, Energy, Commitments	3	4	[3]
P-18	Sprint Planning	Create a content strategy.	Products	3	4	[2]
P-19	Sprint Planning	Decide on a page weight budget.	Products, Carbon Ops	3	4	[4]
P-20	Sprint Planning	Leverage client devices for processing tasks.	Devices, Energy, Utilization	3	4	[1]
P-21	Sprint Planning	Revisit and adjust SLAs and SLOs to align system performance with actual user needs.	Products, Carbon Ops	3	4	[1]
P-22	Sprint Planning	Assess data requirements critically and avoid collecting more data than needed.	Products, Carbon Ops	3	4	[1]
P-23	Sprint Planning	Shrink ML models using techniques like pruning, quantization, compression and distillation.	Products, Carbon Ops, Energy	3	4	[1]
P-24	Sprint Planning	Adopt pretrained ML models and transfer learning to avoid energy-intensive training from scratch.	Products, Carbon Ops, Energy	3	4	[1]
P-25	Sprint Planning	Evaluating the possibility of sending fewer but larger messages between services.	Carbon Ops, Energy	3	3	[1]
P-26	Sprint Planning	When synchronous communication is required, consider using RPCs.	Carbon Ops	3	3	[1]
P-27	Sprint Planning	Adopt serverless architectures that activate resources only when needed.	Utilization, Energy, Carbon Ops	4	5	[1], [4]
P-28	Sprint Planning	When self-hosting, extend hardware lifespan, optimize voltage usage and reduce server idle time.	Utilization, Energy, Devices	4	5	[1]
P-29	Sprint Planning	Do not cause software-defined obsolescence.	Devices, Products	4	5	[1], [2]
P-30	Sprint Planning	Consider using federated learning (FL) for ML model training.	Utilization, Energy, Devices	4	5	[1]

ID	Scrum phase	Measure	GSMM Axes	Maturity Level	Impact Level	Sources
P-31	Sprint Planning	Utilize edge computing for model training.	Utilization, Devices, Energy	4	5	[1]
P-32	Sprint Planning	Use demand shifting to schedule ML training and data collection during periods of low grid carbon intensity.	Energy, Utilization, Carbon Ops	4	5	[1]
E-1	Sprint Execution	Simplify code structure and remove unused code.	Carbon Ops	1	4	[4]
E-2	Sprint Execution	Write clean, standards-based code.	Carbon Ops	1	3	[1], [2]
E-3	Sprint Execution	Use centralized stylesheets and reusable components.	Carbon Ops	1	3	[4]
E-4	Sprint Execution	Follow WCAG accessibility guidelines.	Products	1	2	[2], [4]
E-5	Sprint Execution	Use intentional whitespace between UI elements.	Products	1	1	[4]
E-6	Sprint Execution	Avoid autoplay and use lazy loading techniques.	Carbon Ops, Products	2	5	[3], [4]
E-7	Sprint Execution	Compress images.	Carbon Ops	2	4	[2], [3], [4]
E-8	Sprint Execution	Compress the code before deployment.	Carbon Ops	2	4	[3]
E-9	Sprint Execution	Use video only when it enhances understanding.	Products	2	4	[3], [4]
E-10	Sprint Execution	Use platforms like YouTube for hosting video content.	Utilization	2	4	[2], [4]
E-11	Sprint Execution	Use SVGs for icons and flat color illustrations.	Carbon Ops	2	4	[2], [3]
E-12	Sprint Execution	Reduce HTML complexity and optimize JavaScript DOM handling.	Carbon Ops	2	4	[4]
E-13	Sprint Execution	Compress video and use efficient formats like H.264 or MP4.	Carbon Ops	2	4	[2], [3], [4]
E-14	Sprint Execution	Use efficient image formats like WebP and AVIF.	Carbon Ops	2	4	[3], [4]

ID	Scrum phase	Measure	GSMM Axes	Maturity Level	Impact Level	Sources
E-15	Sprint Execution	Use content patterns and page briefs.	Products	2	3	[2]
E-16	Sprint Execution	Use flat design elements instead of gradients.	Products	2	3	[2]
E-17	Sprint Execution	Use images only when they add meaning.	Products	2	3	[4]
E-18	Sprint Execution	Use system fonts.	Devices	2	3	[2], [3], [4]
E-19	Sprint Execution	Limit the number of fonts to one or two.	Carbon Ops	2	3	[2], [4]
E-20	Sprint Execution	Store pointer references to frequently used DOM elements.	Carbon Ops	2	3	[4]
E-21	Sprint Execution	Block harmful bot traffic by using firewalls.	Utilization	2	3	[3]
E-22	Sprint Execution	Avoid dark UX patterns.	Products	2	2	[2]
E-23	Sprint Execution	Provide printer-friendly versions of web pages.	Products	2	2	[2]
E-24	Sprint Execution	Minimize motion and respect the user's motion media preferences.	Devices	2	2	[3]
E-25	Sprint Execution	Use user-centered testing.	Products	3	4	[2]
E-26	Sprint Execution	Use efficient alternatives to GIFs like WebP or MP4.	Carbon Ops	3	4	[3], [4]
E-27	Sprint Execution	Use appropriate image resolution based on display context.	Carbon Ops	3	4	[4]
E-28	Sprint Execution	Serve images with responsive resolution.	Carbon Ops	3	4	[3], [4]
E-29	Sprint Execution	Optimize custom fonts by converting them to efficient formats like WOFF2 and by subsetting them.	Carbon Ops	3	4	[2], [3], [4]
E-30	Sprint Execution	Structure CSS and JavaScript loading based on criticality.	Carbon Ops	3	4	[4]

ID	Scrum phase	Measure	GSMM Axes	Maturity Level	Impact Level	Sources
E-31	Sprint Execution	Use a "dark mode first" approach.	Devices	3	3	[2], [3], [4]
E-32	Sprint Execution	Use grayscale or monochrome images.	Devices, Products	3	3	[3]
E-33	Sprint Execution	Simplify images and blur unnecessary image details.	Products	3	3	[3], [4]
E-34	Sprint Execution	Combine multiple graphics or images into one file, using CSS sprites.	Carbon Ops	3	3	[2]
E-35	Sprint Execution	Batch DOM changes.	Carbon Ops	3	3	[4]
E-36	Sprint Execution	Construct HTML elements in memory before inserting them into the DOM.	Carbon Ops	3	3	[4]
E-37	Sprint Execution	Inline small, non-SEO-critical images with Base64.	Carbon Ops	3	2	[2]
E-38	Sprint Execution	Embed custom fonts using the CSS @font-face rule or Base64 encoding.	Carbon Ops	3	2	[2]
Rv-1	Sprint Review	Use proxies and accessible tools to estimate the carbon footprint of web products.	Metrics, Footprint	2	3	[1], [3], [4]
Rv-2	Sprint Review	Use right-sized instances and spot instances to reduce overprovisioning in cloud environments.	Utilization, Carbon Ops	3	4	[1]
Rv-3	Sprint Review	Adopt LightSwitchOps to identify and shut down zombie workloads.	Utilization	3	3	[1]
Rv-4	Sprint Review	Use cluster scheduling and containerization to optimize infrastructure-level resource allocation.	Utilization, Carbon Ops	4	5	[1]
Rv-5	Sprint Review	Implement GreenOps.	Utilization, Carbon Ops	4	5	[1]
Rv-6	Sprint Review	Include energy and carbon metrics in Site Reliability Engineering.	Carbon Ops, Metrics	4	5	[1]
Rv-7	Sprint Review	Apply MLOps for efficient and sustainable model management.	Carbon Ops	4	4	[1]
Rt-1	Sprint Retrospective	Encourage reviewing energy-related documentation and guides.	Training	1	2	[5], [6], [7], [8]

ID	Scrum phase	Measure	GSMM Axes	Maturity Level	Impact Level	Sources
Rt-2	Sprint Retrospective	Encourage peer-to-peer learning around green software practices.	Training	2	3	[5], [6], [7], [8]
Rt-3	Sprint Retrospective	Ensure the active involvement of non-technical stakeholders in Sprint Planning, Review and Retrospective sessions.	Commitments, Products	2	3	[9]
Rt-4	Sprint Retrospective	Foster individual accountability and awareness during Sprint Planning and Retrospectives.	Commitments	2	3	[5], [8], [10]
Rt-5	Sprint Retrospective	Use user feedback to detect energy inefficiencies.	Metrics	2	2	[8]
Rt-6	Sprint Retrospective	Consider energy efficiency early in the design and development process.	Carbon Ops, Products	3	5	[8]
Rt-7	Sprint Retrospective	Introduce and experiment with specialized tools to measure or improve energy efficiency.	Carbon Ops, Metrics	3	4	[5], [6], [7], [8]
Rt-8	Sprint Retrospective	Include specific energy efficiency targets in user stories or acceptance criteria.	Metrics, Products	3	4	[8]
Rt-9	Sprint Retrospective	Use profiling tools to detect energy inefficiencies.	Carbon Ops, Metrics	3	4	[8]
Rt-10	Sprint Retrospective	Use static analysis tools for energy/performance profiling.	Carbon Ops, Metrics	3	4	[8]
Rt-11	Sprint Retrospective	Advocate for easy-to-understand, domain-specific sustainability guidelines and policies withing your company.	Commitments, Training	3	4	[9], [11], [12]
Rt-12	Sprint Retrospective	Introduce code reviews focused on identifying energy inefficiencies.	Carbon Ops, Training	3	3	[5], [6], [7], [8]
Rt-13	Sprint Retrospective	Collaborate with educational institutions to promote software sustainability.	Training	3	3	[11], [12]
Rt-14	Sprint Retrospective	Provide internal staff training on sustainability.	Training	3	3	[5]
Rt-15	Sprint Retrospective	Educate customers on the benefits of sustainability.	Training, Products	3	3	[5], [11]
Rt-16	Sprint Retrospective	Invest in energy-efficient refactoring.	Carbon Ops, Products	3	3	[5]
Rt-17	Sprint Retrospective	Mitigate technical debt to allow for sustainable redesign/refactoring.	Carbon Ops, Products	3	3	[5]

ID	Scrum phase	Measure	GSMM Axes	Maturity Level	Impact Level	Sources
Rt-18	Sprint Retrospective	When using software libraries, demand information measure about their energy efficiency.	Metrics, Products	3	3	[5]
Rt-19	Sprint Retrospective	Advocate for legal sustainability standards.	Commitments	4	5	[11]
Rt-20	Sprint Retrospective	Advocate for embedding sustainability in the values and strategy of your organization.	Commitments	4	4	[6], [12]
Rt-21	Sprint Retrospective	When providing software libraries, measure and share energy efficiency information.	Metrics, Products	4	4	[5]
Rt-22	Sprint Retrospective	Advocate for easy-to-understand, domain-specific sustainability guidelines and informative resources outside your company.	Commitments, Training	4	4	[11]
Rt-23	Sprint Retrospective	Advocate for domain-specific sustainability certifications.	Commitments	4	4	[9], [11], [12]
Rt-24	Sprint Retrospective	Develop real-time scenario-aware energy profiling tools.	Carbon Ops, Metrics	5	5	[5], [8]

Bibliography

[1] A. Currie, S. Hsu and S. Bergman, Building Green Software, O'Reilly Media, Inc., 2024.

[2] T. Frick, Designing for Sustainability, O'Reilly Media, Inc., 2016.

[3] T. Greenwood, Sustainable Web Design, A Book Apart, 2021.

[4] M. Andersen, Sustainable web design in 20 Lessons, 2023.

[5] H. Shamshiri, A. Tripathi, S. Oyedeji and J. Porras, "Exploring the Experiences of Experts: Sustainability in Agile Software Development - Insights from the Finnish Software Industry," 2024.

[6] L. Karita, B. C. Mourão and I. Machado, "Software industry awareness on green and sustainable software engineering: a state-of-the-practice survey," in *Proceedings of the XXXIII Brazilian Symposium on Software Engineering*, New York, 2019.

[7] I. Groher and R. Weinreich, "An Interview Study on Sustainability Concerns in Software Development Projects," in *2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, Vienna, 2017.

[8] I. Manotas, C. Bird, R. Zhang, D. Shepherd, C. Jaspan, C. Sadowski, L. Pollock and J. Clause, "An empirical study of practitioners' perspectives on green software engineering," in *Proceedings of the 38th International Conference on Software Engineering*, New York, 2016.

[9] J. García Berná, J. Carrillo de Gea, B. Moros Valle, J. Fernández-Alemán, J. Nicolás Ros and A. Toval, "Surveying the Environmental and Technical Dimensions of Sustainability in Software Development Companies," *Applied Sciences,* vol. VIII, no. 11, p. 2312, 2018.

[10] M. Pilon, "The ‘Planet as a Stakeholder’ Retrospective," The Agile Playground, 10 November 2022. [Online]. Available: https://theagileplayground.com/the-planet-retrospective/. [Accessed 27 May 2025].

[11] S. G. Hansen, H. L. Granum, M. Moen, O. Hulleberg, C. V. Monllaó and Y. Inal, "Current Challenges and Barriers in Sustainable Web Design: A Qualitative Study," in *Proceedings of Mensch und Computer 2023*, New York, 2023.

[12] R. Heldal, N.-T. Nguyen, A. Moreira, P. Lago, L. Duboc, S. Betz, V. C. Coroamă, B. Penzenstadler, J. Porras, R. Capilla, I. Brooks, S. Oyedeji and C. C. Venters, "Sustainability competencies and skills in software engineering: An industry perspective," *Journal of Systems and Software,* vol. CCXI, p. 111978, 2024.