

JOVENES INGENIEROS



INFORME SOBRE

# GREEN SOFTWARE



---

**PREPARADO Y PRESENTADO POR**

HEILAND ABRIL, PUGLIESE LOURDES, MENDEZ AGUSTIN

# INTRODUCCIÓN



El Green Software representa un enfoque estratégico e innovador dentro del campo de las tecnologías de la información, orientado a minimizar el impacto ambiental del desarrollo, despliegue y uso de soluciones digitales. En este informe, se abordará el concepto de Green Software desde diversas dimensiones: qué es y cuál es su naturaleza técnica y ambiental; por qué se ha convertido en una prioridad global frente al crecimiento del consumo energético digital; para qué sirve en los distintos ámbitos empresariales, sociales y tecnológicos; y quiénes son los actores clave involucrados en su implementación, incluyendo desarrolladores, arquitectos de software, empresas y usuarios finales. Asimismo, se analizará cuándo y dónde deben adoptarse prácticas de software sustentable, cómo se aplican en entornos reales, con qué herramientas, métricas y metodologías se cuentan para su desarrollo, y cuánto impacto generan en términos de eficiencia energética y reducción de huella de carbono. A través de este análisis, como empresa de consultoría conformada por ingenieros en sistemas, buscamos no solo entender la relevancia del Green Software, sino también identificar oportunidades concretas para su aplicación en proyectos actuales, aportando al desarrollo tecnológico responsable y sostenible.

# ¿QUÉ ES?

## DEFINICIÓN

El Green Software o software sostenible es aquel que ha sido desarrollado y optimizado con prácticas responsables que consideran el rendimiento energético, la eficiencia de los recursos y la reducción de huella de carbono durante todo su ciclo de vida (diseño, desarrollo, despliegue y uso).

## NATURALEZA

La naturaleza del Green Software es interdisciplinaria y ética. Combina principios de ingeniería de software, eficiencia energética, sostenibilidad ambiental y responsabilidad social. No se limita a programar mejor, sino a tomar decisiones conscientes sobre la arquitectura del sistema, el uso del hardware, el consumo en la nube, y el comportamiento del usuario final.

## SIGNIFICADO

El Green Software significa avanzar hacia una tecnología digital más consciente del entorno. Representa una transformación en la forma en que los ingenieros y las empresas de software piensan su impacto ecológico, incorporando la sostenibilidad como parte fundamental de la calidad del producto y del compromiso social.

## DESCRIPCIÓN

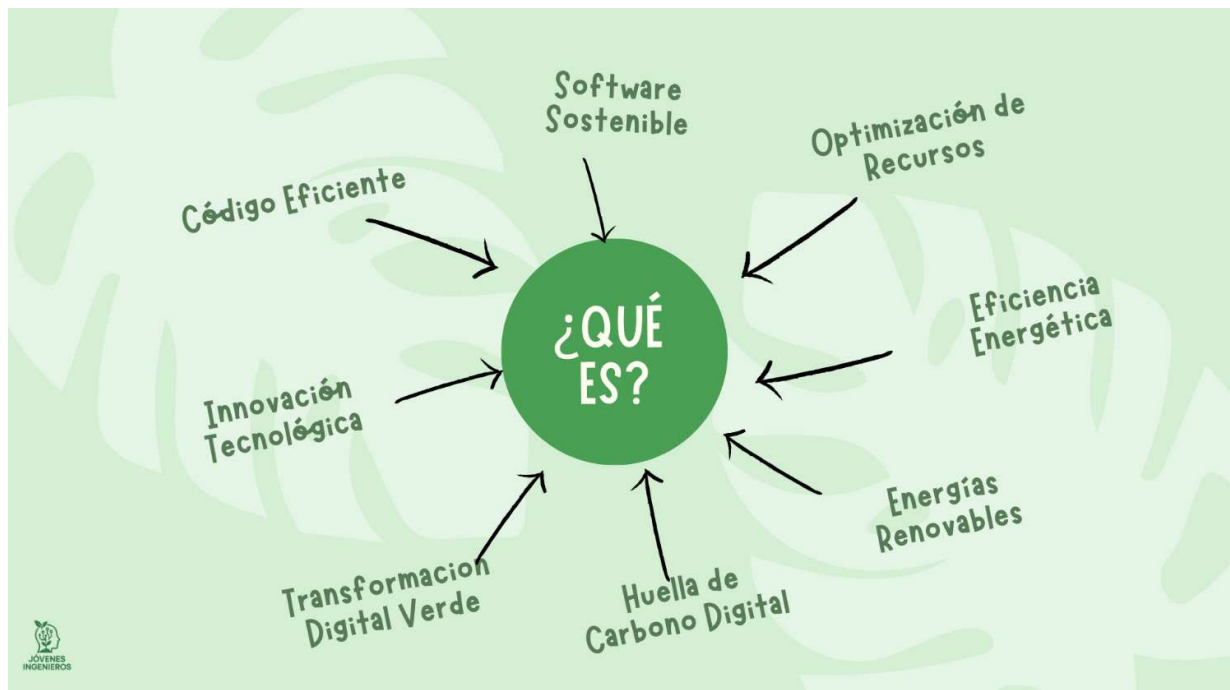
El Green Software promueve:

- Optimización del consumo de energía en procesos computacionales.
- Reducción de emisiones de CO<sub>2</sub> asociadas al uso de servidores y dispositivos.
- Elección de infraestructuras verdes (centros de datos sostenibles).
- Prácticas de codificación eficiente, como disminuir complejidad algorítmica o evitar operaciones innecesarias.
- Conciencia sobre el impacto del software en el hardware (por ejemplo, uso excesivo de la CPU o batería).

También implica medir, rastrear y mejorar continuamente el desempeño ambiental del software en producción

## RESUMEN

El Green Software es una práctica emergente en la ingeniería de sistemas que apunta a reducir el impacto ambiental del software a lo largo de su ciclo de vida. Incluye estrategias de programación eficiente, uso inteligente de recursos, elección de tecnologías sostenibles y evaluación constante de la huella de carbono digital. Esta perspectiva se alinea con los Objetivos de Desarrollo Sostenible y busca combinar innovación tecnológica con responsabilidad ecológica. Estos conceptos se resumen gráficamente en el mapa mental mostrado en la Imagen 1.



**Imagen 1:** Representación visual de los principales ejes del Green Software, destacando diferentes dimensiones.

## CONCLUSIÓN

En un mundo cada vez más digitalizado, el Green Software se posiciona como una necesidad y no solo como una tendencia. Su implementación requiere comprender el software no como un elemento aislado, sino como parte de un sistema complejo que integra componentes técnicos, humanos, ambientales y organizacionales. Adoptar este enfoque implica aplicar una visión sistémica, en la que cada decisión sobre diseño, desarrollo o uso repercute en el funcionamiento del conjunto. Bajo este enfoque, tanto las empresas tecnológicas como los ingenieros en sistemas, CEOs y usuarios individuales pueden asumir un rol activo en la construcción de soluciones digitales sostenibles, demostrando que la eficiencia digital y la sostenibilidad ambiental deben integrarse como partes interdependientes de un mismo sistema.

# ¿POR QUÉ?

## CAUSA

El crecimiento exponencial del uso de tecnologías digitales ha incrementado significativamente el consumo energético y las emisiones de gases de efecto invernadero asociadas a los centros de datos, servidores y dispositivos. La necesidad de minimizar este impacto ambiental es la causa principal para impulsar el Green Software.

## MOTIVO

El motivo es lograr un equilibrio entre el avance tecnológico y la protección del medio ambiente, promoviendo un desarrollo tecnológico responsable que reduzca el consumo energético y la huella de carbono generada por el software y los sistemas digitales.

## RAZÓN

La razón fundamental para adoptar Green Software es mitigar el cambio climático y la contaminación ambiental causada por el sector TI, además de mejorar la eficiencia operativa, reducir costos y cumplir con regulaciones ambientales cada vez más estrictas.

## EXPLICACIÓN

Green Software consiste en diseñar, desarrollar, desplegar y mantener aplicaciones y sistemas de manera que utilicen la menor cantidad posible de recursos energéticos, optimizando el rendimiento sin sacrificar la funcionalidad, y aprovechando tecnologías limpias y sostenibles.

## JUSTIFICACIÓN

El impacto ambiental creciente del sector tecnológico requiere respuestas concretas para limitar el calentamiento global. Adoptar prácticas de Green Software es una medida justificable tanto por la urgencia ambiental como por los beneficios económicos y sociales que conlleva.

## CIRCUNSTANCIAS

La conciencia global sobre el cambio climático, la presión regulatoria, el aumento del consumo digital y la disponibilidad de tecnologías limpias crean un contexto ideal para impulsar el Green Software como una necesidad actual y futura.

## RESUMEN

Green Software se enfoca en reducir el impacto ambiental del sector tecnológico, optimizando el uso de energía y recursos en el desarrollo, implementación y mantenimiento de software. Busca promover prácticas sostenibles que equilibren el avance tecnológico con la protección del medio ambiente, contribuyendo a la reducción de emisiones de carbono y al uso responsable de infraestructuras digitales. Estos conceptos se resumen gráficamente en el mapa mental mostrado en la Imagen 2.



**Imagen 2:** Representación visual de los principales causas del Green Software.

## CONCLUSIÓN

Adoptar Green Software es una acción imprescindible que responde a un sistema de causas y efectos interrelacionados: el aumento del consumo digital, la emergencia climática, las nuevas regulaciones ambientales, y la evolución tecnológica generan impactos que no pueden abordarse de forma aislada. Cada una de estas causas produce efectos directos sobre el medio ambiente y la sociedad, configurando un escenario que exige respuestas integradas. El Green Software se convierte así en una estrategia clave para intervenir en este sistema complejo, reduciendo consecuencias negativas y promoviendo efectos positivos como la eficiencia energética, la innovación responsable y la sostenibilidad a largo plazo.

# ¿PARA QUÉ?

## PROPÓSITO

El Green Software tiene como propósito principal contribuir activamente a la sustentabilidad ambiental a través del diseño, desarrollo y operación de soluciones digitales más eficientes. Este enfoque busca reducir el consumo energético, las emisiones de gases de efecto invernadero y la huella de carbono asociada al ciclo de vida del software. Además, el propósito incluye promover una transformación cultural dentro del sector tecnológico, incorporando la sostenibilidad como eje central en la toma de decisiones y en las prácticas de desarrollo. Al adoptar este tipo de software, las organizaciones no solo mejoran su eficiencia operativa, sino que también fortalecen su compromiso con el cuidado del medio ambiente.

## OBJETIVO

El objetivo específico del Green Software es desarrollar aplicaciones, sistemas y plataformas digitales que funcionen de manera óptima utilizando la menor cantidad posible de recursos computacionales (CPU, memoria, ancho de banda, almacenamiento) y energía. Esto implica implementar prácticas de programación eficientes, elegir arquitecturas sostenibles, reducir el procesamiento innecesario, y fomentar el uso de infraestructura digital de bajo consumo energético. También persigue crear conciencia sobre el impacto ambiental del software en la comunidad tecnológica, incentivando a profesionales y organizaciones a adoptar criterios de sostenibilidad en cada fase del desarrollo: desde el diseño hasta la implementación y el mantenimiento.

## FINALIDAD

La finalidad de este enfoque es lograr una armonía entre la evolución tecnológica y la preservación del planeta. Frente a una creciente demanda de servicios digitales y un incremento sostenido en el consumo energético de los centros de datos, dispositivos móviles y redes, el Green Software se presenta como una alternativa para mitigar los efectos ambientales negativos. Su finalidad es permitir que el avance digital no se dé a costa del deterioro ecológico, sino como una herramienta que acompañe un desarrollo sostenible, asegurando la disponibilidad de recursos para las generaciones futuras.



## MOTIVO

El motivo por el cual surge la necesidad del Green Software responde a varios factores clave: la crisis climática, el aumento de la demanda energética del sector digital, y el crecimiento exponencial de los datos, dispositivos conectados y aplicaciones en la nube. Las tecnologías de la información y la comunicación representan actualmente una parte significativa del consumo energético global. Frente a este escenario, el Green Software nace como una respuesta técnica y ética a la urgencia de reducir el impacto ambiental del ecosistema digital.

## INTENCIÓN

La intención detrás del desarrollo e implementación del Green Software es repensar el rol de la tecnología en la sociedad y cómo esta puede ser parte activa en la construcción de un futuro más sostenible. Se busca cambiar la lógica de desarrollo tecnológico tradicional —que prioriza rendimiento, escalabilidad y rapidez— por una visión integral que también considere el impacto ambiental. Esto implica fomentar la responsabilidad ecológica en los profesionales de la informática, impulsar la innovación sostenible y establecer estándares que favorezcan un uso racional de los recursos tecnológicos.

## RESUMEN

El Green Software es una estrategia orientada a minimizar el impacto ambiental del ecosistema digital, mediante el desarrollo de soluciones tecnológicas que optimicen el uso de energía y recursos. Su implementación implica adoptar principios de eficiencia, simplicidad, reutilización, y reducción del consumo innecesario en el ciclo de vida del software. No se trata solo de un conjunto de prácticas técnicas, sino de una nueva forma de pensar la tecnología en función de su responsabilidad con el entorno. Este enfoque se aplica en múltiples niveles: desde el diseño de aplicaciones y la elección de lenguajes de programación eficientes, hasta la gestión energética en servidores y centros de datos. Estos conceptos se resumen gráficamente en el mapa mental mostrado en la Imagen 3.



**Imagen 3:** Representación visual de las diferentes razones del Green Software.

## CONCLUSIÓN

El Green Software representa una respuesta concreta a los desafíos ambientales actuales, alineando el desarrollo tecnológico con los principios de sostenibilidad. Su adopción permite avanzar hacia una transformación digital responsable, que no solo atienda a la eficiencia funcional, sino también a la eficiencia ecológica. Implementarlo de manera consciente y estratégica es fundamental para reducir el impacto del sector tecnológico sobre el medio ambiente, mejorar la reputación corporativa, y responder a las exigencias de consumidores, reguladores y marcos internacionales. En definitiva, el Green Software no es una opción futura, sino una necesidad actual para construir un modelo de innovación más justo, sostenible y resiliente.

# ¿QUIÉN?

## PERSONA RESPONSABLE

La responsabilidad del Green Software recae principalmente en los equipos de desarrollo tecnológico: diseñadores, programadores, arquitectos de software, y responsables de infraestructura. También son responsables los líderes de proyecto, las empresas tecnológicas, y las organizaciones que impulsan la transformación digital, ya que tienen el poder de tomar decisiones estratégicas sobre qué herramientas, plataformas y prácticas adoptar. Además, los gobiernos y organismos internacionales cumplen un rol esencial al establecer normativas, estándares y políticas que incentiven el uso de tecnologías sostenibles. También debemos tener en cuenta que será el CEO quien debe aprobar las acciones que se propongan, ya que es el encargado de la toma de decisiones en la organización.

## PERSONA INTERVINIENTE

En el proceso intervienen múltiples actores que forman parte del ecosistema digital. Entre ellos se encuentran:

- **DESARROLLADORES:** crean el código con prácticas eficientes.
- **DISEÑADORES DE EXPERIENCIA DE USUARIO (UX/UI):** proponen interfaces más ligeras y funcionales.
- **INGENIEROS DE INFRAESTRUCTURA Y DEVOPS:** gestionan servidores, redes y almacenamiento de manera eficiente.
- **EMPRESAS PROVEEDORAS DE SERVICIOS EN LA NUBE:** ofrecen alternativas energéticamente optimizadas.
- **ACADEMIA E INSTITUCIONES EDUCATIVAS:** forman profesionales con una mirada sostenible.
- **USUARIOS FINALES:** su comportamiento y uso también impactan en el consumo energético

## PERSONA AFECTADA

La falta de Green Software o su implementación inadecuada afecta a:

- EL MEDIO AMBIENTE: por el consumo excesivo de energía, emisiones de CO<sub>2</sub> y agotamiento de recursos.
- LA SOCIEDAD EN GENERAL: debido al aumento de huella digital, uso desmedido de infraestructuras y generación de residuos electrónicos. Futuras generaciones: que heredarán los efectos de un modelo digital insostenible si no se implementan cambios.
- ORGANIZACIONES TECNOLÓGICAS: que pueden enfrentar mayores costos operativos y cuestionamientos éticos

## ROLES

DESARROLLADOR SOSTENIBLE: programa con eficiencia energética en mente.

ARQUITECTO DE SOFTWARE: elige estructuras lógicas y técnicas que reduzcan el uso de recursos.

GERENTE DE PRODUCTO: toma decisiones alineadas con objetivos sostenibles.

ADMINISTRADOR DE SISTEMAS/INFRAESTRUCTURA: optimiza servidores, centros de datos y consumo energético.

USUARIO CONSCIENTE: adapta su comportamiento digital para minimizar impactos.

ENTIDADES REGULATORIAS: fijan marcos de acción y parámetros de evaluación ambiental.

## SISTEMAS INVOLUCRADOS

SISTEMAS OPERATIVOS Y ENTORNOS DE EJECUCIÓN: deben ser eficientes y escalables.

PLATAFORMAS EN LA NUBE (CLOUD COMPUTING): influyen en el consumo energético global.

APLICACIONES WEB Y MÓVILES: optimizables mediante diseño, código y comportamiento del usuario.

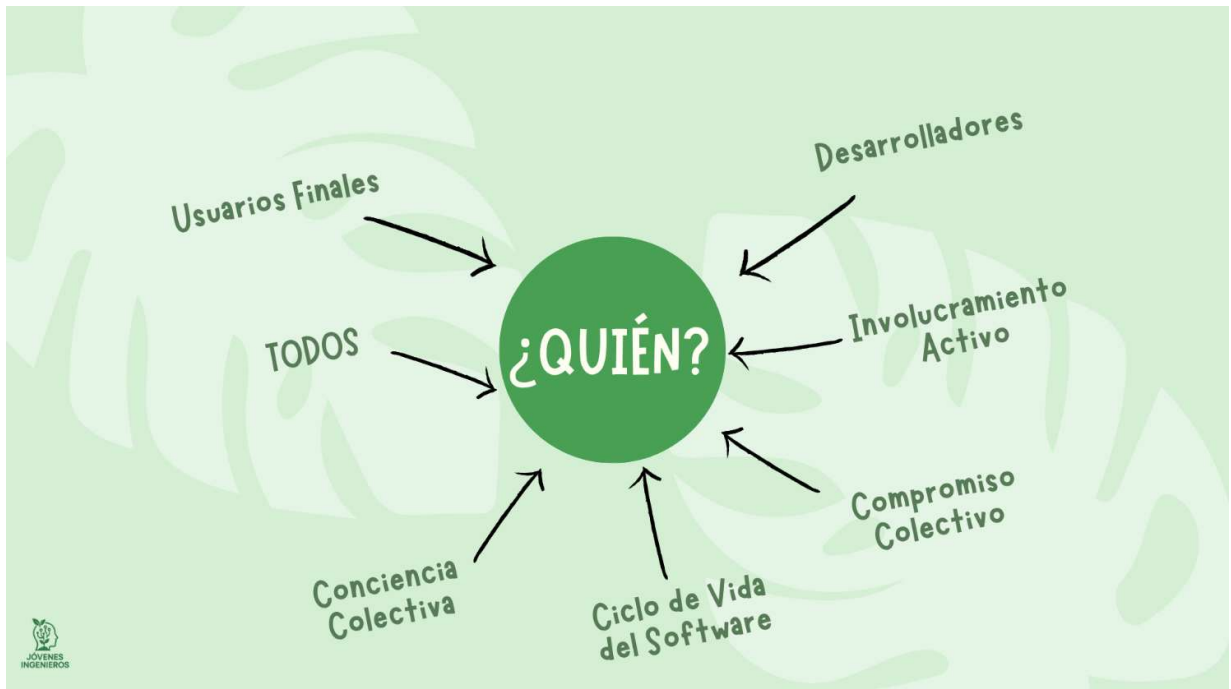
SISTEMAS DE ALMACENAMIENTO Y BASES DE DATOS: su gestión tiene alto impacto ambiental.

CENTROS DE DATOS (DATA CENTERS): actores críticos en la eficiencia energética digital.

SISTEMAS DE MONITOREO: permiten medir y ajustar el consumo en tiempo real

## RESUMEN

El Green Software involucra a una red diversa de personas, roles y sistemas. Desde los equipos técnicos hasta los usuarios finales, todos cumplen una función en el diseño y operación de soluciones digitales sostenibles. La responsabilidad no recae en un único actor, sino que se distribuye a lo largo de todo el ciclo de vida del software, incluyendo a quienes definen políticas, a quienes desarrollan herramientas, y a quienes las utilizan. Esta naturaleza colectiva requiere una coordinación efectiva, educación y compromiso común para lograr un impacto ambiental positivo y duradero. Estos conceptos se resumen gráficamente en el mapa mental mostrado en la Imagen 4.



**Imagen 4:** Representación visual de los principales actores del Green Software.

## CONCLUSIÓN

La pregunta "¿Quién?" aplicada al Green Software permite visibilizar que la sustentabilidad digital es una tarea compartida y transversal. En este sistema de responsabilidades distribuidas, los CEOs ocupan un lugar clave como tomadores de decisiones finales, capaces de establecer políticas organizacionales, asignar recursos y definir el rumbo estratégico hacia una transformación sostenible. Sin embargo, el impacto no depende únicamente del liderazgo ejecutivo: cada individuo, ya sea desarrollador, operador, usuario final o ciudadano digital, tiene la capacidad de contribuir desde su propio rol. La sostenibilidad del software no se logra con acciones aisladas, sino con la participación coordinada de todos los actores del ecosistema tecnológico, desde los niveles más altos de gestión hasta las prácticas cotidianas de uso responsable.

# ¿CUÁNDO?

## MOMENTO

El desarrollo y adopción del *Green Software* debe iniciarse desde las primeras fases del ciclo de vida del sistema. Esto incluye desde el diseño conceptual, pasando por la arquitectura y el desarrollo, hasta el despliegue y mantenimiento. Integrar criterios de eficiencia energética y sostenibilidad desde el inicio garantiza que las decisiones técnicas se orienten a reducir el consumo de recursos a lo largo de toda la operación del software.

## TIEMPO

El tiempo adecuado para implementar estrategias de *Green Software* es ahora. El contexto actual, marcado por la urgencia de mitigar el cambio climático y la creciente demanda energética de infraestructuras digitales, exige un cambio inmediato en la forma en que se conciben y operan los sistemas informáticos. Las tecnologías digitales ya representan un porcentaje significativo de las emisiones globales, por lo que cada segundo de inacción tiene un costo ambiental.

## OCASIÓN

La ocasión más pertinente para aplicar criterios de *Green Software* se presenta en situaciones como la migración a la nube, el rediseño de aplicaciones *legacy*, el desarrollo de nuevas soluciones empresariales o la implementación de arquitecturas escalables. Estas instancias representan oportunidades estratégicas para aplicar principios de eficiencia energética, optimización de recursos y reducción de huella de carbono digital.

## RESUMEN

En términos generales, el *cuándo* del *Green Software* abarca tanto el inicio temprano del diseño como momentos clave de cambio o evolución tecnológica. Debe incorporarse de forma transversal, aprovechando cada ocasión técnica para implementar prácticas sostenibles, con un enfoque proactivo y preventivo en lugar de correctivo. Estos conceptos se resumen gráficamente en el mapa mental mostrado en la Imagen 5.



**Imagen 5:** Representación visual de los momentos para aplicar Green Software.

## CONCLUSIÓN

El Green Software no debe considerarse una etapa aislada, sino un compromiso que atraviesa todo el ciclo de vida del sistema. Su aplicación comienza **antes**, en la fase de diseño y planificación, donde se definen las bases estructurales que determinan la eficiencia futura. Continúa **durante** el desarrollo y despliegue, momento en el que se ejecutan decisiones técnicas clave sobre código, infraestructura y arquitectura. Y debe mantenerse **después**, mediante el monitoreo continuo, la optimización post-lanzamiento y la gestión del fin de vida del software. Esta visión integral, que abarca el antes, durante y después, asegura que cada fase del proceso contribuya activamente a la sostenibilidad digital y a la reducción del impacto ambiental del software.

# ¿DÓNDE?

## SITIO

Los principales sitios donde se manifiesta el impacto del software en términos de consumo energético y huella de carbono incluyen:

- CENTROS DE DATOS (DATA CENTERS): donde se ejecutan las aplicaciones y se almacenan grandes volúmenes de datos.
- INFRAESTRUCTURAS EN LA NUBE (CLOUD COMPUTING): que concentran servicios y procesamiento en servidores distribuidos globalmente.
- EQUIPOS DE USUARIO FINAL (PCS, LAPTOPS, SMARTPHONES): donde el software se ejecuta, afectando directamente al consumo energético individual.

## LUGAR

Los lugares donde se puede aplicar el Green Software van desde:

- EMPRESAS TECNOLÓGICAS: donde se diseñan y desarrollan sistemas.
- ORGANIZACIONES PÚBLICAS Y PRIVADAS: que contratan, desarrollan o consumen software.
- ENTORNOS EDUCATIVOS Y DE INVESTIGACIÓN: que forman a los profesionales con conciencia ambiental.
- AMBIENTES HÍBRIDOS: como espacios de coworking, oficinas remotas y teletrabajo, donde el uso eficiente de recursos tecnológicos también cuenta.

## UBICACIÓN

La ubicación geográfica también influye significativamente:

- PAÍSES CON INFRAESTRUCTURA ENERGÉTICA LIMPIA: ofrecen una ventaja en términos de sostenibilidad.
- REGIONES CON ALTA CONCENTRACIÓN TECNOLÓGICA: donde el impacto ambiental del software puede ser mayor por su escala de operaciones.
- LUGARES CON REGULACIONES AMBIENTALES ERICTAS: que promueven prácticas de desarrollo sostenible.

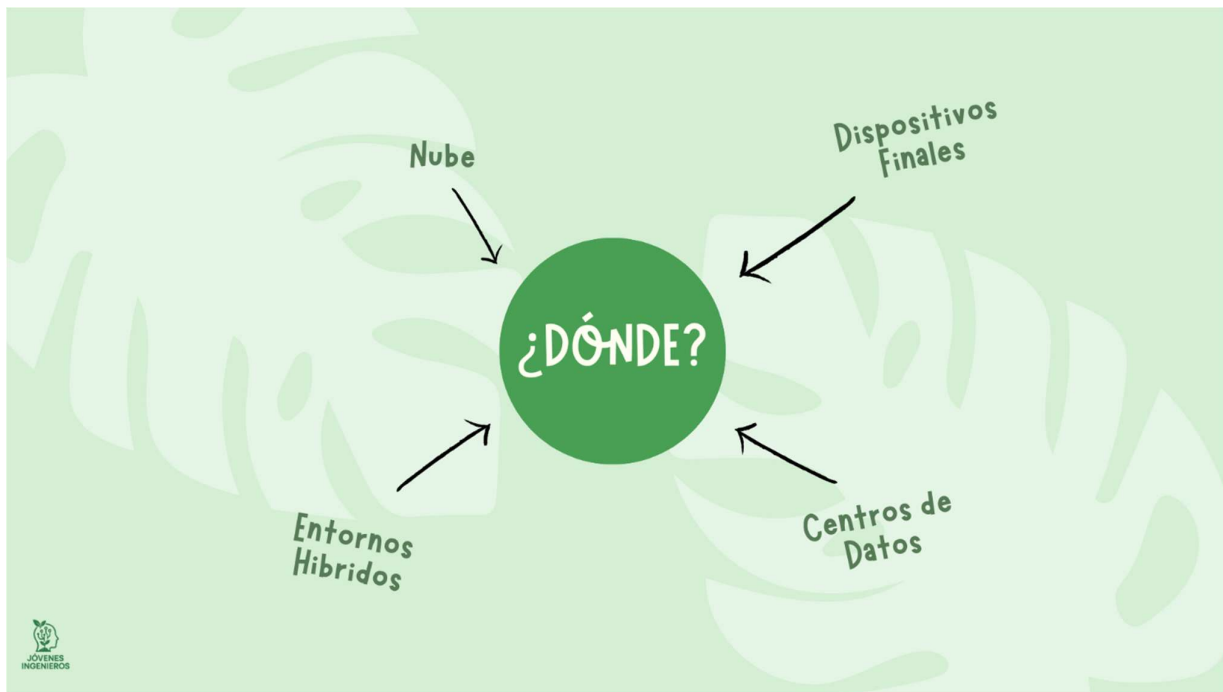


## RESUMEN

El Green Software se manifiesta y actúa en diversos niveles de ubicación:

- Físico (centros de datos, dispositivos).
- Virtual (nubes, entornos digitales).
- Organizacional (lugares donde se toma decisiones sobre el software).
- Geográfico (áreas con más impacto o regulación ambiental)

Estos conceptos se resumen gráficamente en el mapa mental mostrado en la Imagen 6.



**Imagen 6:** Representación visual de los lugares donde se puede aplicar Green Software.

## CONCLUSIÓN

El “dónde” del Green Software es un concepto amplio que abarca tanto espacios físicos como virtuales, locales como globales, en los que se decide, se ejecuta y se consume software. Entender estos lugares es esencial para diseñar estrategias efectivas de sostenibilidad, priorizando aquellas ubicaciones o entornos donde se pueda generar un mayor ahorro energético y reducción de emisiones.

# ¿CÓMO?

## MANERA

El Green Software se implementa de manera consciente y estratégica, priorizando la eficiencia energética y la minimización del impacto ambiental durante todo el ciclo de vida del software. Esto implica tomar decisiones orientadas a la sostenibilidad desde el diseño hasta la operación.

## MÉTODO

### *1. Optimización del código:*

Se busca reducir el consumo de recursos mediante algoritmos más eficientes, estructuras de datos adecuadas y un uso racional del procesamiento. Un código más limpio y liviano implica menor tiempo de ejecución y, por ende, menor gasto energético.

### *2. Desarrollo consciente del ciclo de vida (SDLC verde):*

Incorporar prácticas sostenibles en todas las etapas del ciclo de vida del software. Esto incluye diseño ecoeficiente, pruebas automatizadas con bajo impacto, y mantenimiento orientado a la reducción del consumo energético.

### *3. Uso eficiente de recursos en infraestructura:*

Aplicación de técnicas como *autoscaling*, *serverless computing* o contenedores ligeros, que permiten ajustar dinámicamente la capacidad a la demanda real, evitando el sobredimensionamiento y el uso innecesario de energía.

### *4. Selección de lenguajes y frameworks sostenibles:*

Algunos lenguajes o entornos son más eficientes en términos energéticos que otros. Elegir tecnologías que requieran menor consumo computacional para realizar las mismas tareas puede tener un impacto significativo.

### *5. Estrategias de arquitectura verde:*

Implementación de arquitecturas distribuidas y desacopladas que reduzcan la latencia, optimicen el uso de redes y almacenamiento, y eviten redundancias innecesarias. También se promueve el uso de *edge computing* para reducir la transferencia de datos a grandes distancias.

### *6. Monitoreo y métricas de sostenibilidad:*

Incorporar herramientas que midan la eficiencia energética, huella de carbono digital, y rendimiento sustentable del software. Esto permite establecer indicadores clave (KPIs verdes) para una mejora continua.

### *7. Uso de energías renovables en el hosting:*

Migrar a proveedores de infraestructura en la nube que operen con energía renovable certificada o centros de datos energéticamente eficientes, con políticas claras de sustentabilidad.

## **PROCESO**

El proceso para aplicar principios de Green Software generalmente sigue estas etapas:

1. Diseño sostenible: planificación de sistemas más eficientes desde la arquitectura.
2. Desarrollo responsable: implementación de código optimizado y limpio.
3. Despliegue eficiente: utilización de infraestructura energéticamente optimizada (como servicios cloud sostenibles).
4. Monitoreo y mejora continua: evaluación del consumo energético y ajustes para su reducción.
5. Fin de ciclo: considerar cómo desactivar o reemplazar sistemas obsoletos sin generar residuos tecnológicos innecesarios

## **FORMA**

La forma en que se aplica el Green Software puede variar según el contexto, pero en general se enfoca en:

- Adoptar prácticas de programación eficiente.
- Reducir el uso de recursos computacionales innecesarios.
- Diseñar interfaces simples que reduzcan el procesamiento.
- Implementar procesos de CI/CD que eviten ejecuciones innecesarias.
- Utilizar herramientas de medición del consumo energético

## PROCEDIMIENTO

Un procedimiento común para implementar software verde podría incluir:

- Evaluar la huella de carbono de los sistemas actuales.
- Establecer objetivos de sostenibilidad tecnológica.
- Capacitar al equipo en buenas prácticas.
- Aplicar metodologías ágiles con foco ambiental.
- Documentar las decisiones de impacto ambiental durante el desarrollo

## MODO

El modo de aplicación debe ser transversal e integral: no basta con aplicar una solución puntual, sino que toda la cultura de desarrollo debe alinearse con los principios del Green Software. Esto incluye desde el uso de tecnologías de bajo consumo hasta la concientización de los usuarios sobre el impacto de sus interacciones digitales.

## TÉCNICAS

Entre las técnicas más utilizadas se encuentran:

- Optimización de código: algoritmos más eficientes.
- Uso de lenguajes menos intensivos en recursos.
- Virtualización y contenedores para reducir el consumo de hardware físico.
- Compresión de archivos e imágenes para disminuir el tráfico de red.
- Desactivación automática de servicios inactivos.
- Medición de emisiones de carbono digital con herramientas especializadas

## INSTRUCCIONES

- Establecer objetivos de sostenibilidad digital, claros y medibles.
- Incorporar principios de eficiencia desde el diseño.
- Elegir tecnologías con criterio energético.
- Optimizar el ciclo de desarrollo.
- Implementar prácticas de infraestructura sostenible.
- Medir y monitorear el impacto del software.
- Capacitar al equipo en sostenibilidad digital.
- Auditar y mejorar de manera continua.

## RESUMEN

El “cómo” del Green Software se centra en integrar sostenibilidad en cada etapa del ciclo de vida del software. Desde el diseño hasta la operación, se aplican técnicas y procesos que reducen el consumo energético, mejoran la eficiencia del código y minimizan el uso innecesario de recursos digitales. Estos conceptos se resumen gráficamente en el mapa mental mostrado en la Imagen 7.



**Imagen 7:** Representación visual las maneras de aplicar Green Software.

## CONCLUSIÓN

Aplicar Green Software no es una acción aislada, sino un conjunto de prácticas, procedimientos y decisiones conscientes orientadas a crear soluciones tecnológicas eficientes, responsables y sostenibles. Comprender cómo llevarlo a cabo es clave para cualquier organización que busque alinear su estrategia digital con los objetivos ambientales del presente y del futuro.

# ¿CON QUÉ?

## INSTRUMENTOS

En el desarrollo de Green Software, los instrumentos pueden ser tanto tecnológicos como metodológicos. Algunos ejemplos:

- Herramientas de medición energética (como Green Metrics Tool, Scaphandre, Cloud Carbon Footprint).
- Sistemas de control de versiones que permiten automatizar procesos evitando cargas innecesarias (como Git).
- Plataformas CI/CD optimizadas, que reducen la ejecución redundante de tareas.

## OBJETOS

Aunque el software es intangible, su ejecución depende de objetos físicos, tales como:

- **SERVIDORES Y CENTROS DE DATOS:** donde se aloja y ejecuta el software.
- **DISPOSITIVOS DEL USUARIO FINAL:** desde smartphones hasta computadoras personales.
- **INFRAESTRUCTURA DE RED:** routers, switches, cables, que transportan los datos que procesan las aplicaciones.

## RECURSOS

Los recursos para aplicar Green Software incluyen:

- Energía (idealmente proveniente de fuentes renovables).
- Tiempo de cómputo optimizado.
- Datos e información que permitan tomar decisiones fundamentadas sobre sostenibilidad.
- Capacitación profesional para aplicar buenas prácticas en el ciclo de vida del software.

## ELEMENTOS

Los elementos que componen una estrategia de Green Software incluyen:

- Código limpio y modular.
- Arquitecturas eficientes (como microservicios bien diseñados).
- Sistemas de monitoreo del rendimiento.
- Políticas internas de sostenibilidad tecnológica.

## MEDIOS

Los medios que facilitan la implementación del Green Software pueden ser:

- Infraestructura en la nube con enfoque ecológico (como Amazon EC2 con opción de energía renovable, Microsoft Azure Green, Google Cloud Carbon-Aware).
- Contenedores y virtualización, que permiten ejecutar software con menor uso de recursos.
- Prácticas DevOps sostenibles, que automatizan tareas sin desperdicio computacional.

## HERRAMIENTAS

Algunas herramientas específicas utilizadas en la implementación de software sostenible son:

- Green Software Foundation tools: como el Software Carbon Intensity (SCI).
- CodeCarbon: biblioteca que permite estimar las emisiones de carbono en procesos de ML.
- Ecosia Dev Tools, que promueven el análisis ecológico del desarrollo web.
- Lighthouse (Google): para evaluar la eficiencia energética y de rendimiento de sitios web.

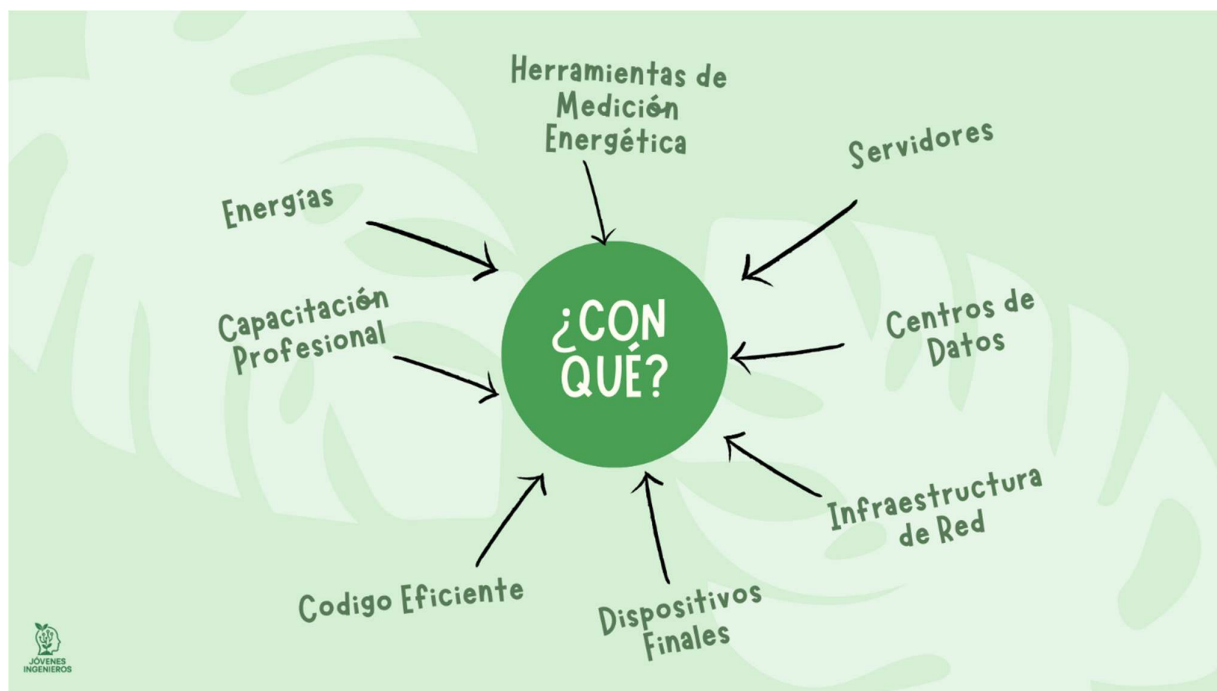
## UTENSILIOS

En este contexto, aunque el término “utensilios” suele aplicarse a entornos físicos, puede adaptarse como una forma más coloquial de referirse a:

- Extensiones o plugins para IDEs que ayudan a identificar código ineficiente.
- Scripts automatizados para medir consumo o limpiar recursos no utilizados.
- Dashboards de monitoreo ecológico para visualización del impacto energético en tiempo real.

## RESUMEN

El Green Software se apoya en una combinación de herramientas tecnológicas, recursos físicos y digitales, medios de infraestructura, instrumentos de medición y políticas organizacionales. Todo esto permite desarrollar soluciones digitales que reduzcan su impacto ambiental. Estos conceptos se resumen gráficamente en el mapa mental mostrado en la Imagen 8.



**Imagen 8:** Representación visual de los elementos que permiten aplicar Green Software.

## CONCLUSIÓN

Comprender con qué se construye el Green Software es fundamental para implementar una estrategia sostenible efectiva y alineada con estándares internacionales. Para ello, es imprescindible contar con instrumentos, recursos técnicos y humanos adecuados, así como adoptar normativas y modelos reconocidos —como el estándar ISO 14001 de gestión ambiental o el modelo Green Software Foundation— que guían las mejores prácticas en todo el ciclo de vida del software. Asimismo, integrar las distintas etapas del desarrollo sostenible —desde el diseño, implementación, pruebas, despliegue hasta la operación y mantenimiento— permite optimizar el uso de recursos, minimizar el impacto ambiental y garantizar la eficiencia energética. De esta forma, la sostenibilidad en el software no solo depende de la intención, sino que se logra mediante el uso consciente de medios concretos, la aplicación de políticas internas, el monitoreo continuo y la mejora basada en métricas ecológicas, asegurando así un compromiso real con el cuidado del medio ambiente.



# ¿CUÁNTO?

## CANTIDAD DE RECURSO

El desarrollo y operación de software requiere una serie de recursos cuya cantidad puede ser medida y optimizada para reducir su impacto ambiental. Por ejemplo:

- Energía eléctrica consumida por servidores durante la ejecución de una aplicación.
- Uso de CPU y memoria en tareas específicas.
- Cantidad de datos transmitidos (que incide directamente en el consumo energético de la red).
- Horas de uso de dispositivos por parte de los usuarios.

## MAGNITUD DEL RECURSO

La magnitud del impacto del software puede medirse en:

- kWh (kilovatios-hora) consumidos por sistemas y servidores.
- Emisiones de CO<sub>2</sub> equivalente (CO<sub>2</sub>e) generadas por el uso de software.
- Petabytes de datos procesados o almacenados por servicios digitales.
- Cantidad de usuarios concurrentes y cómo eso escala el consumo energético.

Ejemplo concreto: se estima que 1 GB de datos transmitido por internet genera entre 3 y 7 gramos de CO<sub>2</sub>e, dependiendo del tipo de red utilizada (WiFi, 4G, fibra óptica).

## NÚMERO

Algunos números ilustrativos del impacto del software digital:

- El sector TIC representa cerca del 3-4% de las emisiones globales de carbono.
- Una videollamada de una hora en HD puede consumir entre 150 MB y 1 GB, dependiendo de la plataforma.
- En promedio, una búsqueda en internet puede emitir entre 0,2 y 0,5 gramos de CO<sub>2</sub>e.

Estos datos permiten visualizar la necesidad de desarrollar software más eficiente y sostenible.

## DETALLES NUMÉRICOS

**REDUCCIÓN ESTIMADA:** Adoptar prácticas de Green Software puede reducir el consumo energético de una aplicación entre 15% y 50%, dependiendo del tipo de optimización.

**AHORRO ENERGÉTICO:** Migrar a una nube con energía renovable puede disminuir la huella de carbono de una empresa tecnológica en un 30-90%.

**IMPACTO POR DISPOSITIVO:** El uso de una app mal optimizada puede consumir un 30% más de batería, lo que implica más frecuencia de carga y mayor consumo eléctrico global.

## DETALLES CUANTITATIVOS

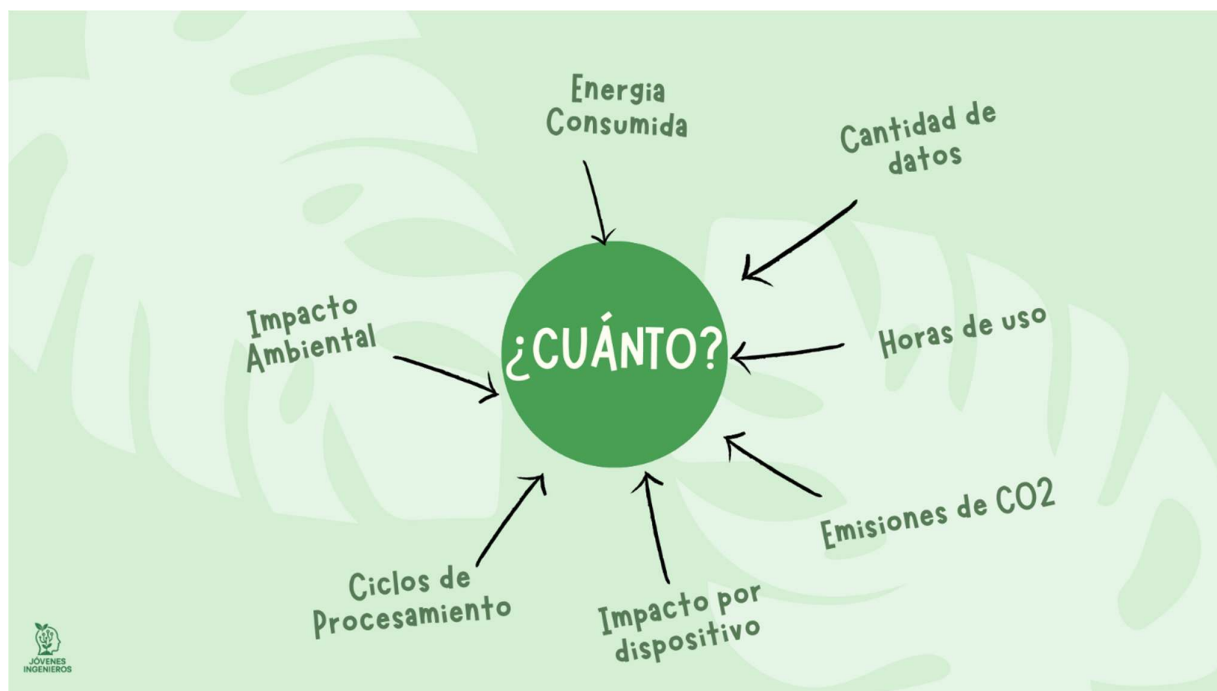
**COMPARACIÓN DE CÓDIGO:** Algoritmos optimizados pueden ejecutar una misma tarea con hasta 70% menos ciclos de CPU que su versión no optimizada.

**ESCALABILIDAD:** En aplicaciones de alto tráfico, una mejora del 10% en eficiencia puede traducirse en una reducción de cientos de megavatios-hora al año.

Estas cifras destacan el valor del enfoque cuantitativo en la planificación de software sostenible.

## RESUMEN

El “cuánto” del Green Software se mide en energía consumida, emisiones generadas, datos transmitidos, ciclos de procesamiento, tiempo de uso y escalabilidad. Estas magnitudes permiten cuantificar el impacto ambiental del software y tomar decisiones orientadas a su reducción. Estos conceptos se resumen gráficamente en el mapa mental mostrado en la Imagen 9.



**Imagen 9:** Representación visual del Green Software desde la perspectiva de recursos implicados.

## CONCLUSIÓN

Saber cuánto consume, contamina o ahorra una solución tecnológica es fundamental para convertir las buenas intenciones en acciones concretas. El enfoque cuantitativo permite identificar cuellos de botella energéticos, establecer indicadores de sostenibilidad y medir el éxito de las estrategias implementadas. En definitiva, lo que no se mide, no se mejora: por eso, el “cuánto” es una pieza clave del Green Software

# CONCLUSIÓN DEL INFORME

A partir del análisis realizado, el conocimiento sobre el Green Software debe transformarse en acciones concretas, estrategias coordinadas y compromiso colectivo. Para ello, es necesario:

1. **Tomar decisiones técnicas más conscientes**, optimizando desde el diseño hasta el despliegue del software para reducir el consumo energético y las emisiones de carbono.
2. **Aplicar prácticas de desarrollo sostenibles**, como escribir código eficiente, elegir arquitecturas verdes y utilizar lenguajes y frameworks que consuman menos recursos.
3. **Contribuir activamente a la sostenibilidad ambiental**, integrando criterios ecológicos en proyectos digitales y alineando el desarrollo tecnológico con los Objetivos de Desarrollo Sostenible (ODS).
4. **Medir y monitorear el impacto ambiental del software**, usando herramientas específicas que permitan evaluar el consumo energético y la huella de carbono digital.
5. **Capacitar y sensibilizar a los equipos de desarrollo**, promoviendo una cultura de sostenibilidad en toda la organización tecnológica.
6. **Impulsar mejoras continuas en infraestructura y procesos**, desde centros de datos más eficientes hasta prácticas de CI/CD que eviten ejecuciones innecesarias.
7. **Aprovechar datos cuantificables para justificar mejoras**, demostrando cómo pequeñas optimizaciones pueden generar grandes ahorros de energía y reducción de emisiones.
8. **Fomentar la colaboración entre actores del ecosistema digital**, desde desarrolladores hasta usuarios finales, reguladores y proveedores de servicios en la nube.
9. **Implementar sostenibilidad desde el inicio del ciclo de vida del software**, para evitar correcciones costosas y maximizar el impacto positivo desde el diseño.
10. **Equilibrar innovación y responsabilidad**, demostrando que el rendimiento tecnológico no está reñido con el cuidado del planeta.

A continuación, se presenta el link de acceso a la pagina web:

<https://abrilheiland03.github.io/jovenesIngenieros/>