

La ética en el empleo del software. Reflexiones sobre la enseñanza universitaria en la postpandemia

Ethical use of software. Considerations about university teaching in the post pandemic context

María Isabel Balmaceda, Ana Laura Cascón-Treo, Mariana del Valle Sánchez-Artola

mariaisabelbalmaceda@educ.ar

Universidad Nacional de San Juan
República Argentina

Resumen. El presente ensayo plantea una reflexión acerca de las implicancias del empleo de software en las prácticas de enseñanza en universidades nacionales argentinas, en el contexto del escenario de educación de emergencia que se configuró a raíz la pandemia y de los que se avizoran en la postpandemia. En la argumentación, por una parte, se analizan el software como producto tecnológico y, por ende, como producto cultural, así como los posicionamientos éticos que conllevan el empleo de software libre y de software privativo. Por otra, se ponen en cuestión patrones de la cultura del consumo que sustenta el mercado del software y las consecuencias de los mismos sobre los usuarios. A modo de ejemplo se exponen datos de casos concretos en la Universidad Nacional de San Juan. Se concluye que, para guardar coherencia con los postulados de la Ley de Educación Superior, ampliar derechos digitales, favorecer la inclusión digital y contribuir a formar profesionales con la autonomía intelectual necesaria para tomar decisiones de orden tecnológico computacional fundadas y hacerse responsables de sus consecuencias, es imprescindible una política de promoción y apoyo del empleo de software libre y del software gratuito en las prácticas universitarias.

Palabras clave: enseñanza universitaria, postpandemia, ética, software

Abstract. This paper aims to debate about the impact of the use of free software in educational activities in the State Universities in Argentina in the emergency scenario due to the pandemic. Consequences that we think are going to remain after this situation. On the one hand, in this essay, the notion of software as a technological product hence as a cultural product is analyzed, and also the ethical implications involved in the use of free software and proprietary software. On the other hand, we inquire about the different cultural patterns implicated in the software industry and their consequences to the users. As an example of that, specific cases, with the Universidad Nacional de San Juan as scenario, are presented. As a conclusion, according to the Higher Education Law (Ley de Educación Superior) the promotion and encouragement to use free software in the educational process in the universities is essential in order to expand digital rights, enhance the digital inclusion and cooperate to educate professionals as individuals with their own opinions, abled to make their own decisions related to the technological and computational options and also responsible for their consequences.

Key words: university teaching, post pandemic, ethics, software

Cómo citar este artículo: Balmaceda, I., Cascón-Treo, A., Sánchez-Artola, M. (2021). La ética en el empleo del software. Reflexiones sobre la enseñanza universitaria en la postpandemia. *Contextos de Educación* 31 (21)

Artículo recibido: 30 de agosto de 2021

Artículo aceptado: 24 de noviembre de 2021

INTRODUCCIÓN

La sorpresa y la incertidumbre derivadas de la situación de pandemia en la que nos encontramos muy repentinamente los docentes universitarios al inicio del ciclo 2020-2021, llevó a que cada quien acudiera, con la mejor buena fe, a los recursos más a mano o más conocidos, para cumplir con su rol de la mejor manera posible. En muchos casos, ignorando las implicancias de estas acciones. Los docentes tuvimos que adaptarnos y la interfaz educativa (Scolari, 2018), tuvo que evolucionar. Implementamos cambios que hacía tiempo veníamos intuyendo como necesarios. En pocos meses vivimos una transformación digital que pensábamos, llevaría años. Transitamos un camino que no podemos desandar en la postpandemia y que hoy nos exige cuestionarnos prácticas docentes, en muchos casos, naturalizadas en la presencialidad.

Habiendo transcurridos 18 meses de educación no presencial, entendemos que la inocencia está perdida. A partir de la experiencia del dictado de clases a distancia, sabemos ahora que el empleo de cualquier software privativo que permita almacenar datos en la *nube* puede implicar el riesgo de robo o uso indebido de datos de los estudiantes, además de *hackeos* o *malware*, cuando exige instalación de programas. Sabemos ahora, que la metáfora de *la nube* oculta (no ingenuamente) que se trata de un servidor físico alojado en algún país y administrado por alguien que no conocemos (Stallman, 2007). También comprendimos, desde la propia experiencia, que el empleo de software privativo, sin respetar las condiciones de su licencia, nos colocaba en la ilegalidad e impedía el necesario respaldo institucional para las prácticas docentes a distancia.

En este sentido consideramos que las acciones de la Universidad Nacional de San Juan (UNSJ), a partir del trabajo del equipo que integra su SIED (Sistema Institucional de Educación a Distancia), son un ejemplo a seguir. Esta institución en el transcurso de tres meses desde la suspensión de las actividades presenciales, en 2020, puso a disposición de sus docentes tres plataformas: una de aulas virtuales, una de salas de videoconferencias y una de almacenamiento en la nube, que se sumaron a una de blogs, existente desde hacía algunos años atrás. La característica que comparten estas cuatro plataformas es que se montaron sobre softwares libres (*Moodle*, *Big Blue Button*, *NextCloud* y *WordPress*). Esto ha permitido garantizar que no se hace uso indebido de los datos de los estudiantes y los docentes (porque permanecen guardados en servidores de la UNSJ). Además, durante más de un año y, a raíz del convenio del Gobierno Nacional con las empresas prestadoras de servicios de telefonía digital, permitió la conexión sin consumo de datos móviles para cientos de estudiantes y docentes. En la actualidad las plataformas mencionadas siguen siendo personalizadas y ajustadas periódicamente por profesionales de la institución, para adaptarlas a las necesidades de docentes y estudiantes que las emplean a diario.

Pero el caso descrito no responde a una política integral de la UNSJ, sino solo del equipo que gestiona el Sistema de Educación a Distancia. Al interior de cada unidad académica no existen lineamientos que orienten en determinada dirección a los docentes. El software y su empleo no se han problematizado hasta ahora, ni tampoco sus categorías de distribución. Esto sumado a que, desde el sentido común, las opciones de software privativo aparecen como las más fáciles y accesibles, ha contribuido a naturalizar situaciones de empleo ilegal de programas tanto para el dictado de clases como para la ejercitación por parte de los estudiantes.

Fundamentaremos a continuación, por qué pensamos que la discusión acerca del uso ilegal de software privativo versus el empleo de software libre/software legal debe darse en el seno de las universidades nacionales. Para ello previamente, desarrollaremos algunos conceptos y expondremos algunos datos que, entendemos, resultan centrales en la configuración de los escenarios actuales de enseñanza en el nivel universitario.

EL SOFTWARE

Diego Levis dice acerca del software que “La esencia de la máquina es el software, pero la esencia del software es su filosofía” (2009, p. 232). Las computadoras permiten al usuario ejecutar las instrucciones que los programadores incluyeron en ellos, y así son éstas las que condicionan y regulan qué es lo que se puede o no hacer con el software. Es decir que en cierto modo el software modela las actividades y

hábitos de los usuarios. En este sentido Manovich sostiene que “...el software informático naturaliza el modelo de autoría entendido como selección de librerías de objetos predefinidos” (2005, p.184). Lo que implica que siempre que operamos una computadora y por ende, cada vez que usamos un programa, lo hacemos a través de una serie de filtros culturales plasmados en la interfaz y en la lógica con que se organizaron los datos digitales. Como contraparte, cuando utilizamos los programas y empleamos las operaciones que estos ofrecen, es decir las técnicas propias que incluyen “...éstas se convierten en parte de cómo nos entendemos, a nosotros mismos, a los demás y al mundo. Las estrategias de trabajo con datos informáticos se vuelven nuestras estrategias cognitivas de carácter general” (Manovich, 2005, p.171).

EL SOFTWARE LIBRE Y EL SOFTWARE PROPIETARIO

Desde el sentido común se suele asumir que el software libre es aquel que puede descargarse desde la web sin costo alguno. No obstante, el software libre puede tener un costo y el software propietario (o privativo) puede ser gratuito. No es la gratuidad lo que distingue, en primer término, a estas dos categorías de programas, sino la posibilidad de acceso a su código fuente. En el software libre el código fuente (la programación) no está encriptado y por lo tanto cualquier persona con conocimiento del lenguaje de programación en el que está escrito, puede acceder a él, modificarlo para introducir mejoras y publicarlo nuevamente con la misma particularidad. Otra de las características del software libre es el posicionamiento filosófico en la base de su desarrollo. Recordemos que el Movimiento del Software Libre a través de la *Free Software Foundation* (FSF) defiende que la humanidad debe tener el derecho de controlar los programas que usa y todo lo que estos hacen. Para que eso sea así, entiende que es necesario que los usuarios puedan ejercer cuatro libertades:

- La libertad de ejecutar el programa como se desee, con cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y cambiarlo para que haga lo que se desee (libertad 1). El acceso al código fuente es una condición necesaria para ello.
- La libertad de redistribuir copias para ayudar a otros (libertad 2).
- La libertad de distribuir copias de sus versiones modificadas a terceros (libertad 3). Esto le permite ofrecer a toda la comunidad la oportunidad de beneficiarse de las modificaciones. El acceso al código fuente es una condición necesaria para ello.

(*Free Software Foundation*, 2021-A)

Lo público del código permite que exista una comunidad en muchos casos formada no solo por informáticos profesionales, sino también por educadores, especialistas en tecnología educativa, en *e-learning*, diseñadores, etc., según el caso, que lo depuran y lo mejoran continuamente y una comunidad de usuarios que se benefician de este trabajo y que pueden, o no, hacer aportes económicos. En suma, en la metodología típica de desarrollo del software libre, permanentemente y de forma iterativa se proponen, debaten, diseñan, desarrollan y prueban nuevas funcionalidades. Todo este proceso se funda en un modelo de servicio público y de solidaridad (Adell y Bernabé, 2007) en el que se priorizan valores como la libertad de pensamiento y expresión y el beneficio de la sociedad, por encima del interés particular, todos coincidentes con los que sustentan a la educación pública en nuestro país.

Por el contrario, lo que encontramos en la base filosófica del software propietario es, usualmente, el espíritu de lucro. Es por eso que el código fuente (la programación) del software propietario siempre está encriptada, guardada con celo como secreto empresarial, de modo tal que solo puede ser modificado por su desarrollador. Este hecho no constituye en sí mismo un defecto, pero sí conlleva una serie de aspectos negativos desde el punto de vista que nos interesa. El primero de ellos es que, dado que el usuario no tiene posibilidad de conocer la programación, en ella se pueden incluir diferentes *DRM*¹. Estas son defendidas por los fabricantes de software, que sostienen que se trata de un sistema de protección de archivos sujetos a derecho de autor, propio del software. Pero que, a juicio de la FSF (2020), con quien acordamos, constituye un sistema que permite incluir instrucciones en el código fuente de un programa privativo, que pueden restringir ilegalmente lo que los usuarios pueden hacer con los medios

digitales. Estas restricciones pueden involucrar, entre otras cosas, la inserción de rutinas que espíen comportamientos digitales del usuario y los transmiten para reunirlos en una base de datos (violando el derecho a la privacidad), la imposición de empleo de determinado hardware (cuando se incluyen instrucciones que impiden el funcionamiento del programa a menos que se cuente con determinados recursos y/o marcas de hardware), y en casos extremos, la tergiversación de datos, ya sea por un error involuntario en el código o premeditadamente.

Otro aspecto negativo que trae aparejado el software propietario es que está sujeto al devenir del mercado. Es decir, su continuidad en el tiempo no tiene necesariamente que ver con los requerimientos de los usuarios, sino con decisiones de política empresarial. Tomemos como ejemplo lo que ocurrió con Flash hace unos años atrás. Este software, originalmente propiedad de la empresa Macromedia, en los primeros años de la década de 2000 se había convertido en líder en el mercado de programas de animación y fue ampliamente utilizado para dar interactividad a páginas web. No obstante, a partir de su compra por parte de la empresa Adobe, en 2005 y por conflictos de estricta política empresarial, en pocos años sus productos dejaron de ser reconocidos por los navegadores web. Esto provocó que miles de páginas web que incluían productos de Flash dejaran de funcionar. De este modo Flash, un programa muy potente con una interfaz de lo más intuitiva, que ofrecía excelente relación tiempo de aprendizaje/logros, fue condenado a su desaparición por decisiones absolutamente ajenas a los usuarios que habían pagado por una licencia. Es decir, a raíz de intereses corporativos, la comunidad de usuarios debió aceptar que el tiempo invertido en la construcción de un sitio web, fuera tiempo perdido. Asimismo, y de forma equivalente, lo fue el tiempo invertido en estudiar y aprender a usar el programa. Todo esto sin que mediara un ínfimo pedido de disculpas a los usuarios que alguna vez compramos la licencia del programa.

Otra particularidad del software propietario, no directamente relacionada con la encriptación de su código, pero muy relevante para nosotros, tiene que ver con las *demandas de hardware* y software para poder funcionar. Las empresas de software suelen publicar versiones nuevas de sus programas periódicamente. Estas versiones siempre dicen ofrecer ventajas cualitativas respecto a las anteriores y, por regla general, demandan más recursos de hardware que sus predecesoras (Véase, a modo de ejemplo, la Tabla 1). Esto no sería tan grave si no existieran políticas coordinadas con los fabricantes de sistemas operativos, porque podríamos pensar en continuar empleando una versión antigua de un programa si, para nosotros como usuarios, esta fuera funcional. Pero ocurre que Microsoft, la empresa que fabrica el sistema operativo más difundido en nuestra comunidad de usuarios, nos impele a migrar a versiones nuevas de ellos, mediante su política de dejar de ofrecer soporte de actualizaciones de seguridad para las versiones antiguas. Es decir, nos extorsiona descaradamente para que elijamos entre actualizar la versión de Windows, o quedar expuestos a *cyber* ataques. Entonces si elegimos actualizar la versión de Windows tenemos altas probabilidades de que nuestros antiguos softwares y algún hardware, dejen de funcionar correctamente. Sin mencionar que, si la nueva versión de Windows no es compatible con nuestros recursos de sistema, necesitaremos adquirir nuevo hardware. Esto evidencia una de las formas en las que el mercado nos impone patrones de consumo irracionales, como si fueran naturales, limitando nuestra posibilidad de elección a la de elegir qué compramos, en lugar de poder elegir si compramos, o no. Este es solo un ejemplo de los efectos colaterales de lo que Bauman (2007) denomina, *cultura del consumo* en la que la libertad del sujeto es entendida como libertad de elegir y consumir (Posadas Velázquez, 2013).

Se genera de este modo una espiral perversa que está en la raíz de procesos de exclusión crecientes. Cuantos más recursos de hardware sean necesarios para instalar programas para cursar a distancia, a menos estudiantes podremos integrar. Si le sumamos los costos del software privativo, la situación se agrava aún más.

	Adobe Photoshop CS3	Adobe Photoshop CS6	Adobe Photoshop CC
Sistema operativo	Windows® XP con Service Pack 2	Windows® XP con Service Pack 3 o Windows 7 con Service Pack 1. Compatibles con Windows 8.	Windows 10 (64 bits) versión 1809 o posterior; las versiones LTSC no son compatibles
Procesador	Pentium 4 , Centrino o Xeon	Intel® Pentium® 4 o AMD Athlon® de 64 bits	Intel® o AMD con compatibilidad de 64 bits; 2 GHz o más rápido con SSE 4.2 o posterior
RAM	512MB	1 GB	8 GB
Espacio en Disco	650MB	1 GB	4 GB
Monitor	Resolución de 1024x768	Resolución de 1024 x 768 (1280 x 800 recomendada)	Resolución de 1280 x 800
Placa de video	64 Mb de VRAM	512 MB de VRAM	4 GB de memoria de GPU

Fuente: Elaboración propia

Tabla 1. Comparación de requisitos mínimos de sistema para diferentes versiones del programa Adobe Photoshop

En el software libre las demandas de hardware son, en la generalidad de los casos, muy bajas. Sumado a ello, las actualizaciones de versiones, que también son muy frecuentes, no suelen implicar mayores exigencias de equipos respecto a las previas. Veamos un ejemplo de ello en la Tabla 2, donde comparamos los requerimientos de hardware de las últimas versiones de softwares de edición de imágenes, de ilustración y de maquetación editorial (categorías de software usualmente empleadas en el dictado de carreras de Diseño Gráfico y de Comunicación) equivalentes en software libre y propietario. Los requerimientos de los programas privativos cuadruplican y en algunos casos decuplican los de sus equivalentes en software libre.

Requerimientos	Software libre	Software privativo o propietario
	INKSCAPE 1.1	ILLUSTRATOR CC
Procesador	Pentium, 2,2 GHz	Intel multinúcleo (con soporte de 64-bits) o AMD Athlon 64
Memoria RAM	2 Gb de RAM	8 GB (16 GB recomendado)
Espacio libre en disco	200 MB	2GB
Sistema operativo	GNU/Linux, Mac y Windows	Windows 10 (64-bit) versiones 1809, 1903, 1909, 2004.
	GIMP 2.10.12	PHOTOSHOP CC
Procesador	Intel Pentium IV	Intel de sexta generación o superior o equivalente de AMD
Memoria RAM	256 MB	8 GB
Espacio libre en disco	236MB	7,3 GB; se requiere espacio libre adicional durante la instalación
Sistema operativo	Sistema operativo: GNU/Linux, Mac, Windows, Unix y otros.	Microsoft Windows 10,64 bits, versión 1903 o posterior (versión 1909 recomendada)
	SCRIBUS 1.4.x	INDESIGN CC
Procesador	Pentium, 2,4 GHz	Intel multinúcleo (con soporte de 64-bits) o AMD Athlon 64
Memoria RAM	2 Gb de RAM	4GB (recomendados 16 GB)
Espacio libre en disco	1 GB	3,6 GB + espacio adicional
Sistema operativo	GNU/Linux, Unix, Mac OS X y Windows.	Windows 10 (64-bit) versiones 1607, 1809, 1903, 1909, 2004.

Fuente: Elaboración propia

Tabla 2. Comparación de los requisitos de sistema necesarios para instalar algunos programas del paquete Adobe, con los de sus equivalentes en software libre

Otro factor muy importante a analizar es *el costo*. Los programas detallados en la Tabla 2 como equivalentes a los del Paquete Adobe, son gratuitos y lo van a ser siempre (como gran parte del software libre). No están sujetos a políticas del mercado, sino que dependen de la comunidad de programadores y usuarios que le dan soporte. Asimismo, existe profuso material bibliográfico en formato de documentos de texto y videotutoriales, disponibles en la web, de manera gratuita, para aprender a usarlos. Este hecho beneficia ampliamente a las comunidades educativas de instituciones públicas, en primer término, porque no requieren hacer erogaciones para equipar las computadoras de los gabinetes o salas, con programas legales y tampoco obliga a los estudiantes a pagar licencias para instalarlos en sus computadoras. (Una licencia individual, para un solo equipo del paquete Adobe cuesta hoy alrededor de \$38.000 -unos U\$D370- anuales.) Aun cuando una institución puede gestionar un paquete de licencias educativas, para el caso de los programas listados, no son gratuitos y también requieren de un pago anual relevante. En segundo término, porque nos permite a todos (estudiantes y docentes) trabajar en un marco de legalidad y en igualdad de condiciones.

Si bien, los ejemplos de los programas que hemos comparado suelen ser primordialmente empleados en Carreras de Diseño Gráfico o de Comunicación, podríamos hacer similares comparaciones con algunos

programas que se emplean en otras. De hecho, el enfoque que se sustenta en este escrito es aplicado desde hace tres ciclos, en la implementación del Requisito de Informática que incluye el Plan de Estudio de la Carrera de Arquitectura y Urbanismo de la Facultad de Arquitectura, Urbanismo y Diseño (FAUD - UNSJ). A partir de un proyecto que fue elaborado por dos de las autoras de este escrito y aprobado por el Consejo Directivo de la FAUD en 2018, el Curso de Apoyo para rendir la prueba de Suficiencia en informática, estipulada en el currículo, se dicta con programas que se emplean legalmente, ya sea por ser software libre (LibreCad y Open Office) o porque la empresa que lo produce ofrece una versión gratuita para su empleo en línea (SketchUp). Al momento del examen, el Tribunal examinador da libertad a los estudiantes para elegir los programas con los que desarrollarán las tareas solicitadas, solo deben cumplir con la entrega de archivos en determinados formatos, en un cierto tiempo.

Por último y desde lo pedagógico, se entiende que la implementación de softwares libres puede ser útil para promover la participación de los estudiantes y de la comunidad educativa en la construcción de conocimiento y en el desarrollo de nuevos programas. Reforzando la idea del software libre como una construcción colectiva, se podría incentivar a los estudiantes a ser usuarios críticos, activos, que reporten posibles fallas o problemas y puedan sugerir cambios a las comunidades de desarrolladores y aun integrarlas (Adell y Bernabé, 2007). La FSF (2021b), afirma: “El primer programa libre que alguien usa o instala, incluso en un sistema operativo propietario, puede llevarte a un viaje de por vida”. Entendemos que, en el contexto actual, es un viaje que vale la pena emprender.

LA OPINIÓN DE LOS ESTUDIANTES

Exponemos aquí los resultados de una encuesta realizada a estudiantes de dos asignaturas de la carrera de Diseño Gráfico de la FAUD UNSJ, que durante la pandemia cursaron empleando software libre. Aun cuando sabemos que la muestra abordada no es representativa del total de los estudiantes de la FAUD UNSJ, consideramos que sí lo es del universo de quienes realizaron una experiencia de aprendizaje empleado software libre durante este período.

En las asignaturas mencionadas, el software se utiliza para llevar adelante ejercicios exploratorios a través de los cuales los estudiantes desarrollan su capacidad de interactuar con los medios digitales para lograr algún propósito preestablecido. Las tareas (fotomontajes, vectorización de un logo-isotipo, composición de una maqueta editorial, animación de una marca, entre otras) en todos los casos, tienen relación con los trabajos que les son solicitados en otras asignaturas de la carrera y demandan dar respuesta a problemas de orden tecnológico computacional desde programas de edición de imágenes, de ilustración, de maquetado editorial y/o de animación en 3d.

A través de la encuesta se indagó acerca de la opinión de los estudiantes respecto a su experiencia y a su sentir a futuro en relación con el software libre. La misma fue respondida de manera anónima y la pregunta semiabierta “¿con qué tipo de software preferirías continuar trabajando el resto de tu carrera?” incluía tres opciones de respuesta:

- Preferiría seguir trabajando con software libre el resto de mi carrera
- Preferiría seguir trabajando con software propietario el resto de mi carrera
- Preferiría combinar ambas opciones
- ¿Por qué? (respuesta opcional)

En los gráficos 1,2 y 3, a continuación, se muestran los resultados obtenidos.

Como puede verse, en el Gráfico 1, en 2020 algo más de la mitad de los estudiantes encuestados, que estaban en segundo año de la carrera, (habiendo cursado el primero presencial) dijeron preferir seguir empleando software libre el resto de su carrera. Nueve de ellos afirmaron preferir trabajar con software propietario y tres, combinar opciones. En 2021, solo respondieron la encuesta 28 estudiantes de los 51 que completaron el cursado (Gráfico 2). La proporción de ellos que respondió que preferiría trabajar con software libre fue aún más elevada que el año anterior (dieciséis estudiantes), y solo ocho de los estudiantes que respondieron manifestaron preferir trabajar con software propietario.

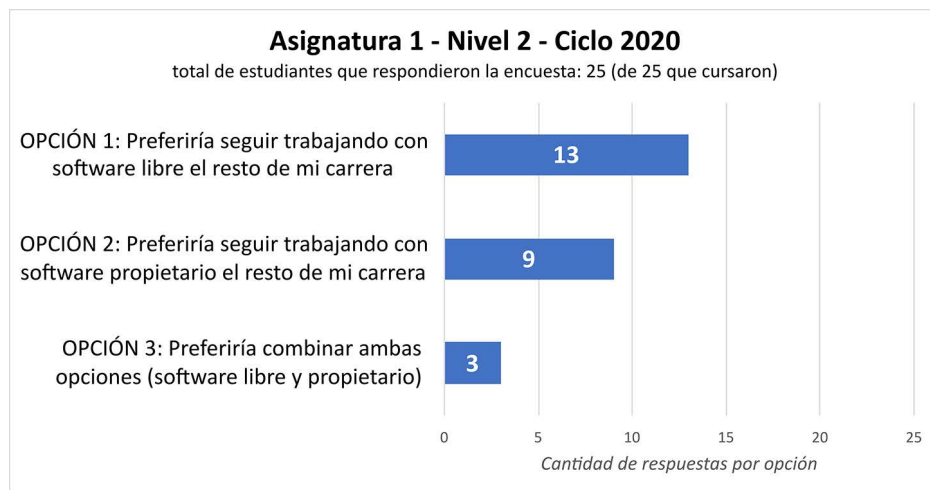


Gráfico 1. Opinión de estudiantes que cursaron a distancia la Asignatura 1 de la Carrera de Diseño Gráfico de la FAUD-UNSJ en 2020

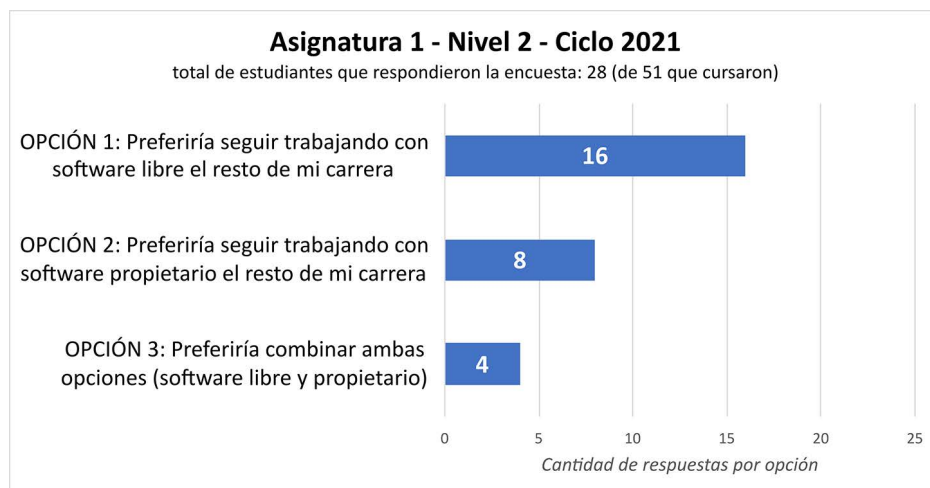


Gráfico 2. Opinión de estudiantes que cursaron a distancia la asignatura 1 de la Carrera de Diseño Gráfico de la FAUD-UNSJ en 2021

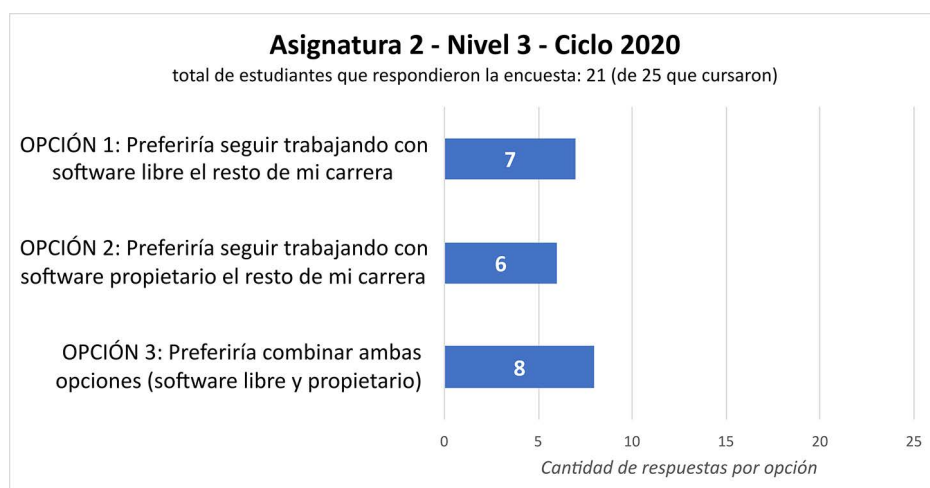


Gráfico 3. Opinión de estudiantes que cursaron a distancia la asignatura 2 de la Carrera de Diseño Gráfico de la FAUD-UNSJ en 2020

En el grupo de estudiantes encuestados del tercer nivel de la carrera, que en 2020 cursaron a distancia la Asignatura 2, pero que habían cursado al menos dos años de manera presencial (Gráfico 3), siete dijeron preferir emplear software libre y ocho, combinar las dos opciones. Solo seis de estos sujetos se expresaron a favor de emplear software propietario.

Podemos decir que gran parte de estos estudiantes, a partir de una experiencia concreta, ven como posibilidad cierta el empleo de software libre o la combinación de opciones.

Entre los fundamentos que expusieron los estudiantes que dijeron preferir el software propietario, elegimos algunas frases para ilustrar lo que, entendemos es un problema del que como docentes, somos en parte responsables y debemos hacernos cargo.

Estudiante 1	<i>Me parecen importantes ambos softwares [libres y propietarios], si bien, al no tener los recursos para adquirir legalmente los softwares propietarios para trabajar con ellos, son lamentablemente los que más nos solicitan dentro de la carrera y en trabajos. Éticamente me parece correcto trabajar con software libre los cuales también son de alta utilidad, pero prácticamente prefiero el software propietario por lo explicado anteriormente.</i>
Estudiante 2	<i>La ética de uso de los softwares nunca está de más saberla pero no es algo esencial en nuestra carrera, ni algo aplicable en otras materias por lo que me agradaría que o se redujera ese contenido, o en su defecto que se diera</i>
Estudiante 3	<i>...en otras materias de la carrera nos exigen utilizar un programa de adobe.</i>
Estudiante 4	<i>...¿por qué enseñar programas como Gimp o Inkscape sabiendo que Illustrator y Photoshop son los programas que se usan con más frecuencia?</i>
Estudiante 5	<i>Gimp e Inkscape son interesantes y muy buenos, pero no son tan fundamentales como Illustrator y Photoshop de Adobe</i>
Estudiante 6	<i>Son útiles y de fácil acceso, [Gimp e Inkscape] pero al no ser el estándar que se utiliza en la mayoría de los trabajos, con colegas, resulta difícil poder usarlos ya que en el trabajo se comparten archivos originales y se editan entre todos, y la mayoría tiene los softwares propietarios, por ende, resulta inconveniente utilizarlos.</i>

Se puede observar que, en la mayoría de los comentarios hay una naturalización respecto a lo que se *debe usar* aludiendo a que es lo que todos usan, lo que se exige o lo que es importante. Incluso un estudiante cuestiona que se pongan en juego aspectos éticos en el empleo del software, porque la *ética no sería aplicable en su carrera* (algo que, si bien no es generalizado, sin duda debe alarmarnos). En ninguno de los casos hay reflexión, solo aceptación de una realidad que consideran inmodificable y producto de una cierta lectura de su experiencia en la universidad.

Esta situación evidencia que se requiere mayor claridad en los mensajes relacionados con el mundo del software, desde nuestra institución y también coherencia en ellos con los propósitos de la educación en las universidades nacionales argentinas y los principios enunciados en los planes de estudio.

CONCLUSIÓN

La suspensión de actividades presenciales a raíz de la pandemia por COVID 19 nos enfrentó con una realidad que en la presencialidad se mantenía velada: la mayor parte de nuestros estudiantes (y muchos de nuestros docentes) no se encuentran en condiciones de invertir en la compra de software, en el mejor de los casos sí, en condiciones de invertir en hardware. Es decir que el costo del software privativo sumado a las demandas de hardware de última generación que este exige, configuran un escenario en el que la educación a distancia, desde una universidad pública, no habría sido posible, al menos en el marco de la legalidad. En este contexto el trabajo con software libre se reveló como la mejor y única alternativa.

En otras palabras, la crisis provocada por la suspensión de la presencialidad puso en evidencia algo que como institución pública deberíamos haber considerado mucho antes, la necesidad imperiosa de guardar coherencia con lo que establece la Ley de Educación Superior "...garantizar la igualdad de oportunidades y condiciones en el acceso, la permanencia, la graduación y el egreso..." (Ley de Educación Superior 24.521 Art.2) de los estudiantes. En el contexto actual y en el que avizoramos como de postpandemia, esto solo se puede lograr promoviendo el trabajo con software libre y gratuito facilitando así la inclusión digital.

El menosprecio de muchos docentes universitarios sobre softwares libres como *Inkscape*, *Gimp*, *Scribus*, *Blender* o muchos otros, tiene sus raíces en un profundo desconocimiento, no solo de su potencial sino también de los fundamentos en los que se sustentan. Desconocimiento que es explotado por el mercado del software privativo que, con poderosas campañas de difusión, contribuye a fomentar el consumo acrítico de la tecnología sin instancias de evaluación de las opciones posibles y de las consecuencias que traen aparejadas. Esto es, la promoción de lo que Prieto Castillo (2004) denomina *analfabetismo tecnológico*. Si como institución avalamos y estimulamos el consumo irracional de tecnología, estamos validando la selección de nuestros estudiantes de acuerdo con su capacidad de consumo y su poder adquisitivo.

Por otra parte, si en nuestras clases ejercitamos empleando software libre y no software privativo, esto no va a modificar la capacidad de diseñar o producir del egresado, dado que la computadora solo es capaz de reproducir lo que el ser humano instancia en ella (Benbenaste, 1999). En cambio, sí va a incidir en el posicionamiento ético en el que nos situamos como docentes y en el que ubicamos al estudiante. Por lo tanto, sí va a condicionar la persona y el profesional que contribuimos a formar.

En este sentido, a esta altura del siglo, consideramos que no podemos soslayar que la tecnología en general y la tecnología computacional como subconjunto de ella, son productos culturales y, por lo tanto, no neutrales. En su diseño están implícitos tanto la cosmovisión como los intereses de los actores que participaron en él (Giuliano, 2008). Es por ello que "...las herramientas que utilizamos moldean nuestra forma de vida en las sociedades modernas donde la técnica se ha generalizado" (Feenberg, 1999, p. 2). Entonces la elección o el apoyo de una tecnología constituye "...una decisión ontológica plena de consecuencias políticas" (Feenberg, 2000, p.22). Porque cualquier sistema tecnológico limita y condiciona las formas posibles de organización social (Fourez, 2005).

En particular los medios digitales, en muchos casos, como en el del software privativo, "...sirven para reforzar el tecnocapitalismo ayudando a la integración empresarial a nivel mundial en detrimento del poder de los estados" (Levis 2021, p.1), Los estados ven así reducida su capacidad de intervenir en este campo para reducir brechas y promover la equidad.

Por lo expuesto consideramos que, en el momento actual, como institución del Estado, podemos elegir estar a la altura de las circunstancias históricas para intentar modificarlas o plegarnos ante su fuerza arrolladora. Bauman (2010) sostiene que uno de los propósitos de las instituciones estatales es promover visiones alternativas a la del consumismo como forma de evitar la exclusión y la marginación. Contribuir a que primen los sentimientos de solidaridad y la responsabilidad ética. Entendemos que es nuestra obligación como miembros de una universidad pública no solo encontrar las formas de promover el empleo de software libre, sino también garantizar que no se presione a los estudiantes para usar software

privativo. De ningún modo podemos avalar el ubicarlos en la disyuntiva de pagar por licencias, o cometer un delito empleando programas ilegalmente para cursar una carrera universitaria.

Entendemos que este es el camino si lo que perseguimos es una ampliación de derechos, una inclusión digital y fundamentalmente la formación de profesionales con la autonomía intelectual necesaria para tomar decisiones de orden tecnológico computacional fundadas y hacerse responsables de sus consecuencias. De otro modo nos estaremos limitando a formar a solo unos pocos trabajadores eficaces, funcionales al mercado, individuos como productos deseables y atractivos y estaremos aceptando como daños colaterales (Bauman, 2013) la exclusión de la educación universitaria de muchos.

Notas

1. La sigla DRM para los fabricantes de software significa *digital rights management* (gestión de derechos digitales). En tanto para la FSF, su significado en los hechos es *digital restrictions management* (gestión de restricciones digitales).

Referencias

- Adell, J. y Bernabé, Y. (2007). Software libre en educación. Disponible en https://www.researchgate.net/publication/216393192_Software_libre_en_educacion
- Bauman, Z. (2007). *Vida de consumo*. México: Fondo de Cultura Económica.
- Bauman, Z. (2010). *Modernidad Líquida*. México: Fondo de Cultura Económica.
- Benbenaste, N. (1995) *Sujeto = Política x Tecnología / Mercado*. Buenos Aires: Oficina de Publicaciones del C.B.C. - UBA.
- Feenberg, A. (1999). *Questioning Technology*. Londres: Routledge.
- Feenberg, A (2000). *Transformar la tecnología: Una nueva visita a la teoría crítica*. Buenos Aires: Universidad Nacional de Quilmes Editorial.
- Fourez, G. (2005). *Alfabetización científica y Tecnológica*. Buenos Aires: Colihue.
- FSF (2020). Defective by design Campaign. Consultado en https://www.defectivebydesign.org/so_youve_got_some_questions_do_you#examples
- FSF (2021a) ¿Qué es el software libre? Consultado en <https://www.gnu.org/philosophy/free-sw.html>
- FSF (2021b). The journey begins with a single step: climb the freedom Ladder. Consultado en: <https://www.fsf.org/blogs/community/the-journey-begins-with-a-single-step-climb-the-freedom-ladder>
- Giuliano, G. (2008). Tecnología, desarrollo y democracia: hacia otra artificialidad posible. *Scientiae Studia*, vol. 6 (3). Disponible en: Scielo Brasil: <http://dx.doi.org/10.1590/S1678-31662008000300006>
- Levis, D. (2009). *La pantalla ubicua*. Buenos Aires: La Crujía.
- Levis, D. (2021). Pensar la educación en la sociedad de la pantalla. Consultado en: <https://tecnoeducativas.wordpress.com/2021/04/14/6o-encuentro-20-4-21-material-de-apoyo-y-lecturas/>
- Manovich, L. (2005). *El lenguaje de los nuevos medios de comunicación*. Barcelona: Paidós.
- Ley Nº 24.521 de 1995. Ley de educación superior. Agosto de 1995. Modificación. Ley 27204 de 2015. Ley de implementación efectiva de la responsabilidad del Estado en el nivel de educación superior. Noviembre de 2015. Publicada en el Boletín Oficial de la República Argentina. Disponible en

<https://www.boletinoficial.gob.ar/detalleAviso/primera/11496684/null>

Posadas Velázquez, R. (2013). La vida de consumo o la vida social que se consume: apreciaciones sobre la tipología ideal del consumismo de Zygmunt Bauman. *Estudios Políticos*, vol. 9, (29), pp. 115-127.

Prieto Castillo, D. (2004). *La comunicación EN la educación*. 2da Edición 2009, Buenos Aires: La Crujía.

Scolari, C. A. (2018). *Leyes de la interfaz: Diseño, Ecología, Evolución, Tecnología*. Barcelona: Editorial Gedisa.

Stallman, R. (2009). Entrevista en vivo en Canal 7, Televisión Pública Argentina el 31 de enero de 2009 por la periodista Rosario Lufrano.

Artículo recibido: 30 de agosto de 2021

Artículo aceptado: 24 de noviembre de 2021