



UNIVERSIDAD DE CIENCIAS Y ARTES DE CHIAPAS

FACULTAD DE MÚSICA

Uso del software libre como alternativa para la creación e investigación de la música en la Facultad de Música de la UNICACH.

QUE PARA OBTENER EL TÍTULO DE

LICENCIADO EN JAZZ Y MÚSICA POPULAR

PRESENTA

PEDRO MAGAÑA LEÓN

ASESOR: MTRO. LUIS AUGUSTO CANTORAL GONZÁLEZ

Tuxtla Gutiérrez, Chiapas

Octubre del 2023.





UNIVERSIDAD DE CIENCIAS Y ARTES DE CHIAPAS

2

SECRETARÍA GENERAL

DIRECCIÓN DE SERVICIOS ESCOLARES

DEPARTAMENTO DE CERTIFICACIÓN ESCOLAR

AUTORIZACIÓN DE IMPRESIÓN

Lugar: Tuxtla Gutiérrez, Chiapas

Fecha: 18 de octubre 2023

C. Pedro Magaña León

Pasante del Programa Educativo de: Licenciatura en Jazz y Música Popular

Realizado el análisis y revisión correspondiente a su trabajo recepcional denominado:

Uso del software libre como alternativa para la creación e investigación de la música en la Facultad de

Música de la UNICACH

En la modalidad de: Elaboración de texto

Nos permitimos hacer de su conocimiento que esta Comisión Revisora considera que dicho documento reúne los requisitos y méritos necesarios para que proceda a la impresión correspondiente, y de esta manera se encuentre en condiciones de proceder con el trámite que le permita sustentar su Examen Profesional.

ATENTAMENTE

Revisores

Mtro. Alexis Fabián Tovilla Martínez

Lic. Félix Rodríguez León

Mtro. Luis Augusto Cantoral González

Firmas:

Ccp. Expediente

Contenido

Introducción.....	10
Capítulo I. La Evolución de la Escritura Musical del Siglo V al Siglo XXI	
Historia de la Escritura Musical.....	13
Edad Media.....	13
Ars Nova.....	14
Renacimiento.....	14
Barroco.....	14
La Influencia de la Imprenta en la Caligrafía Musical.....	16
Programas de Edición y Producción Musical.....	18
Principales Software de Edición de Partituras.....	19
Encore.....	19
Sibelius.....	19
Finale.....	20
Lilypond.....	20
Guitar Pro.....	21
MuseScore.....	21
Estaciones de Trabajo de Audio Digital (DAW).....	21
Pro Tools.....	22
FL Studio.....	22
Ableton Live.....	23
Audacity.....	23
Ardour.....	24
Reaper.....	24

	4
Studio One.....	25
Sistemas Operativos.....	26
Mac OS X.....	26
Windows.....	27
GNU/Linux.....	27
De los Instrumentos Físicos a los Controladores e Interfaces MIDI.....	29
Capítulo II. Diferencias Entre Software Libre y Software Privado	
Tipos de Licencias en la Industria del Software y la Música.....	34
Licencia de Software Propietario.....	35
Licencia Perpetua y Suscripción.....	36
Licencia de Dominio Público.....	38
Licencia Permisiva.....	38
Licencias Restrictivas.....	39
Licencia LGPL.....	39
Licencia Pública General (GNU).....	40
Licencias Creative Commons.....	41
Capítulo III. Presentación de los Software de Producción Musical.....	
MuseScore como Propuesta de Editor de Partituras.....	46
Ardour: Propuesta de Estación de Trabajo de Audio Digital (DAW).....	52
Consideraciones Generales.....	55
Capítulo IV. Implementación de Ubuntu Studio para uso Académico	
¿Por qué la implementación de esta Plataforma?.....	57
La Compatibilidad de las Interfaces de Audio con el Kernel de Linux.....	58
Protocolo de Comunicación y Puertos E/S.....	61

	5
FireWire.....	61
Thunderbolt.....	62
USB (Universal Serial Bus).....	62
Formato SoundFont.....	63
SoundFont.....	64
Emulación de Instrumentos Virtuales, Formatos Comerciales y de Código Abierto.....	64
Virtual Studio Technology (VST).....	65
Audio Units (AU).....	65
AAX.....	66
Formato de plug-ins para GNU/Linux.....	66
LADSPA.....	66
DSSI.....	66
LV2.....	67
SF2/3 y SFZ.....	67
Sin Limitaciones.....	67
Capítulo V. Pruebas de Rendimiento de Ubuntu Studio en Diferentes Gamas de Computadoras	
Ubuntu Studio.....	69
Equipos y Hardware Externo de Prueba.....	70
Equipos de Gama Baja.....	71
Equipo de Gama Media.....	72
Interfaces de Audio Externas.....	73
Uso del Hardware y Software Propuesto para la Producción Musical.....	74
Apartado Gráfico de PulseAudio y ALSA.....	74
QjackCtl.....	77

	6
Carla.....	79
Muscore.....	82
Ardour.....	86
Sonic Visualiser.....	92
Lo Importante del Software sin Limitaciones.....	93
Conclusiones.....	94
Referencias.....	96

Índice de figuras

Figura 1. Información de la página oficial de descargas de MuseScore.....	47
Figura 2. Presentación de la pantalla de inicio de MuseScore.....	48
Figura 3. <i>Presentación de la pantalla Centro de Inicio</i>	48
Figura 4. <i>Área principal de trabajo</i>	50
Figura 5. <i>Presentación del área de trabajo, con la transcripción de un arreglo para Big Band</i> ..	51
Figura 6. <i>Presentación de la página web de Ardour</i>	53
Figura 7. Ventana de inicio de Ardour.....	53
Figura 8. <i>Área de trabajo de Ardour</i>	54
Figura 9. <i>Área de mezcla y mucho más</i>	55
Figura 10. <i>Presentación de la página oficial de Ubuntu Studio</i>	69
Figura 11. <i>Preferencias del sistema, Audio</i>	75
Figura 12. Gestión del hardware y los perfiles en la barra de tareas del escritorio.....	76
Figura 13. <i>Gestión y configuración del audio en cada aplicación</i>	76
Figura 14. <i>Presentación de QjackCtl en funcionamiento</i>	77
Figura 15. <i>Configuración de QjackCtl</i>	78
Figura 16. <i>Patchbay gráfico</i>	79
Figura 17. <i>Pestaña de Rack en Carla</i>	80
Figura 18. <i>Presentación de la pestaña de Patchbay de Carla</i>	81
Figura 19. <i>Patchbay de Carla</i>	82
Figura 20. <i>Parámetros de configuración Musescore</i>	83
Figura 21. <i>Musescore bajo el control de QjackCtl</i>	84
Figura 22. <i>Mezclador de Musescore</i>	85
Figura 23. <i>Línea de tiempo</i>	86

Figura 24. Área de trabajo de ardour.....	87
Figura 25. Área de Mezcla.....	87
Figura 26. Ventana de ruteo para señales de audio y MIDI.....	88
Figura 27. Como se ven los ruteos del DAW en Carla.....	89
Figura 28. Lista de plugins.....	90
Figura 29. Lista de plugins (continuación).....	91
Figura 30. Presentación de Sonic Visualiser.....	92
Figura 31. Analizador de espectro audible.....	92
Figura 32. Espectrograma de rango melódico.....	93

Índice de tablas

Tabla 1. <i>Clasificación de los logos que conforman las distintas licencias Creative Commons...</i>	42
Tabla 2. <i>Tipos de licencia oficiales por Creative Commons.....</i>	43
Tabla 3. <i>Computadora de gama baja.....</i>	71
Tabla 4. <i>Computadora de gama media.....</i>	72
Tabla 5. <i>Interfaces de audio probadas.....</i>	73

Introducción

Interactuar con las computadoras para la investigación y creación de la música, en ocasiones suele dar problemas como usuarios y durante un largo tiempo estarán presentes; pero ciertas situaciones podrían resolverse haciendo uso del software libre.

Un ejemplo de este tipo de problemas se da en la comunidad universitaria que generalmente depende del uso de software privado, y por situaciones económicas o de otra índole no se puede adquirir para poder trabajar las tareas y actividades. Todo esto surge a raíz del mal uso que el usuario le da al software, como instalar parches ilegales (no confundir con los parches creados por la empresa que distribuye el software original) que se aplican usualmente copiando uno o varios archivos pequeños a la carpeta del software específico; o también aplicando un *crack*, que son pequeñas aplicaciones que se ejecutan, quedando activas en segundo plano y que el usuario no puede ver, ni controlar, pero le permite liberar las herramientas de trabajo que son necesarias para investigar, crear música o compartir el conocimiento. El peligro de estas técnicas es que también se aprovechan para poder copiar la identidad de una persona o robar información de manera remota o automatizada de una computadora donde fue instalado el crack o parche e incluso hasta llegar a bloquear el hardware o dañarlo permanentemente. De igual forma los software portable usan las mismas técnicas y no olvidemos que los software ilegales infringen las leyes y puede causar sanciones legales contra empresas e individuos.

El mal uso que un usuario le da al software trae problemas con el uso del hardware tanto a la hora de consolidar un proyecto creativo, como también en aspectos legales a la hora de querer publicar nuestros proyectos y monetizarlos. Hoy en día la industria de la música se ha actualizado para hacer uso del internet y poder distribuir este arte de una manera más rápida haciendo que las plataformas de *streaming* como YouTube, Spotify, Deezer, Apple

Music y otras más, lograrán cambiar el mercado de la música y pasaran de vender un producto final a brindar un servicio por suscripción. Posteriormente, estas mismas reglas fueron adoptadas por las grandes empresas de software, en especial las que se dedican a la producción musical, pudiendo llegar, incluso, a rastrear sus licencias mediante el escaneo de los metadatos de los archivos creados cuando se envían a una empresa de distribución.

En el presente proyecto se plantea la utilización del software libre para el estudio y la producción musical en la Facultad de Música de la Universidad de Ciencias y Artes de Chiapas (UNICACH), logrando que los estudiantes tengan acceso a estas tecnologías que hoy en día, más que una innovación, son herramientas esenciales ya que el mundo cada día está más interconectado gracias a la internet, así como en la música los artistas, en muchas ocasiones, colaboran con otros por pequeños o casi nulos incentivos para ganar reconocimiento en la industria. Es aquí donde el software libre puede ayudar a la institución, al brindarle todas las herramientas que los estudiantes necesitan para su aprendizaje, en vez de obligar al estudiante a comprar licencias cuyo costos son generalmente elevados u optar por parches que, en términos legales, a futuro podría representar un problema.

De esta manera y a través del uso de software libre podemos lograr unas de las resoluciones colaborativas e individuales más grandes de la historia, promoviendo así la libertad en la creación y el conocimiento mediante el uso de la tecnología.

En el capítulo uno se habla sobre los rasgos más importantes de la escritura musical mostrando su evolución hasta nuestros días, el cómo se adaptó a las nuevas tecnologías y las personas que participaron en dichos avances tecnológicos.

En el capítulo dos se muestra una comparativa y las principales diferencias entre el software libre y el software privado. Se expone, además, la importancia de la protección de todo trabajo artístico, junto con los procesos legales pertinentes para que este no sea robado

por una persona que no sea su legítimo creador y las consecuencia que conlleva el hacer mal uso de las herramientas u obras artísticas.

En el capítulo tres se presentan los software propuestos, su entorno gráfico y algunas utilidades relevantes para la producción musical, como lo son MuseScore y Ardour, entre otros.

En el capítulo cuatro se explica la importancia del uso de software libre, bajo qué especificaciones deben ejecutarse, los equipos de cómputo en los que se pueden instalar, proponiendo estrategias para la implementación de “Ubuntu Studio” como sistema operativo para su aplicación en el entorno académico.

Por último, en el capítulo cinco se revisan los diversos software incorporados en el sistema operativo “Ubuntu Studio” para mostrar su apartado visual, bajo qué condiciones se pueden usar y cómo funcionan y se comportan según el rendimiento observado en los equipos de cómputo de prueba.

Con esta investigación sobre las herramientas que ofrece el software libre para la creación e investigación de la música, encontramos alternativas de nivel profesional, que brindan todas las herramientas necesarias para obtener trabajos de calidad, sin restricciones como en plataformas privadas, de igual forma los desarrolladores permiten que personas o instituciones apoyen en el mantenimiento de todo el ecosistema del software libre, lo cual favorece que esta alternativa esté disponible para cualquier interesado ya sean estudiantes o profesionales de la música.

Capítulo I. La Evolución de la Escritura Musical del Siglo V al Siglo XXI

Historia de la Escritura Musical

Para poder entender cómo la tecnología revolucionó la escritura musical nos tenemos que remontar a la Edad Media donde por siglos se desarrollaron métodos y procesos de enseñanza y escritura para preservar las obras musicales.

Edad Media

“A partir de la Edad Media, se atribuye tradicionalmente al papa Gregorio Magno una serie de reformas e innovaciones concernientes a la música eclesiástica que encontraron su culmen en lo que se conoce como canto gregoriano” (Maymó, 2013, p.150.)

Hacia el siglo IX, Hucblado de Saint Amand (840-930), propuso la escritura de las notas en el viejo sistema de Beocio, que consistía en letras sobre un pautaado de seis líneas correspondientes a las cuerdas de la cítara. A principios del siglo XI, apareció el tetragrama, con líneas de colores que indicaban la altura de los *neumas* y así el cantor podría leerlos. Con posterioridad a Guido. Tiempo después, el tetragrama perdió su colorido y las líneas empezaron a ser dibujadas sencillamente en monocromo. (Dominguez y Rigueiro, 2015, p. 12.)

Guido De Arezzo (991 d.C.-1050 d.C.) retomó el uso de la línea roja años atrás, a la que agregó una amarilla arriba y un par de negras por arriba y debajo de la roja, creando un intervalo de terceras (nota - espacio - nota). Con esto, pretendía que los alumnos también reconocieran a primera vista cómo sonaría esa nota de sólo leer el escrito y reconocer los intervalos entre cada una de las figuras representadas. Usó su mano como un recordatorio de las notas, en la que cada dedo podía representar una nota sola o conjunto de ellas, lo que se popularizó como la “mano guidoniana”.

Las notas siguieron escribiéndose en escritura neumática cuadrada hasta que a partir del siglo V fueron creándose las figuras de las notas y las claves tal como las conocemos. (Dominguez y Rigueiro, 2015, p. 14.)

Ars Nova

Con el desarrollo del Ars Nova (siglo XIV), se buscó la perfección y estética musical en los efectos sonoros. También se retomó la notación rítmica, sistema mensural de Franco de Colonia. “La nueva notación extiende la sistematización a los valores más breves y a todas las divisiones y subdivisiones posibles. Esto proporcionó a los compositores herramientas muy potentes para hacer una música con una complejidad rítmica inaudita en la música” (González, 2021, p. 28.)

Renacimiento

Durante el Renacimiento en el siglo XIV, se reutilizaron diversas características que permitieron a los músicos establecer técnicas de composición más precisas.

Una de las más importantes a resaltar fue que los músicos ya disponían de partituras y se desarrolla una relación más estrecha entre música y letra. Como ya se ha comentado anteriormente, la estructura de la obra vendrá definida en la mayoría de los casos por la del texto literario. Además, se aprecia el intento de hacer el texto comprensible y de expresar con la música los conceptos y sentimientos sugeridos por el texto. (González, 2021, pp. 33 – 34.)

Barroco

En el periodo Barroco (finales del siglo XVI), la música tomó una importancia más fuerte ya que se desarrollaron diversas técnicas, como el bajo continuo, normalmente abreviado como “b. c.”

En la música de esta época el bajo tiene como función de definir la armonía, el

esquema básico de toda composición, su perfil estructural, va a ser el de melodía y b. c. Lo demás es un complemento que, muchas veces, no se escribe y será el intérprete del bajo continuo quien lo complete a discreción y de manera improvisada. (González, 2021, p. 47.)

Otra de estas técnicas fue la armonía como resultado del contrapunto y el bajo figurado que es una representación visual de esas armonías comúnmente empleadas en música.

Con el bajo figurado, los números, las alteraciones o los símbolos se colocaron sobre la línea de bajo que fue leída por los intérpretes de instrumentos de teclado, como los intérpretes de instrumentos de teclado, como el clave o el órgano, y de otros instrumentos como el laúd. Los números, las alteraciones o los símbolos indican al teclista los intervalos que debe tocar sobre cada nota de bajo. El teclista improvisaría un acorde de voz para cada nota de bajo. (Música barroca, s. f, párr. 10.)

El uso de texturas, ornamentación y dinámicas (fuertes contrastes sonoros), fue una parte crucial de la música barroca.

Los instrumentos predominantes son los instrumentos de cuerda frotada, cuya relación con la utilización de las texturas bipolares y permitió ofrecer claridad y transparencia en el sonido. Aparte, se da el uso de instrumentos de viento como la flauta y el oboe como instrumento solista. En cuanto a instrumentos con capacidad de acordes tenemos el arpa, órgano, clave y laúd, entre otros. (Rivera. A. S, 2020, párr. 6, 14, 15.)

El desarrollo de la tonalidad está basado en el uso de dos tipos de escalas, en modo mayor y menor (básicamente una de cada una, aunque con variantes) las que organizan la música. Es a partir de este momento cuando comienzan a surgir obras con una estructura formal más complejas, como concierto en Do mayor o Sonata en Re menor y se dice que esa es la tonalidad en la que están escritas (Do mayor y Re

menor). (González, 2021, p. 48.)

En el barroco habían dos maneras de tratar el ritmo, podríamos decir opuestas:

1) Ritmo declamatorio, libre, sin medida, regido por una flexibilidad extra musical. Es el ritmo del recitativo afectivo y el de algunas piezas improvisatorias. Es un gran logro en la representación más fiel de los afectos. 2) Ritmo mecánico, regular, medido, con pulsaciones mecánicamente recurrentes. Junto a estos se dan todas las posibilidades intermedias y frecuentemente se usan contrastados, por ejemplo, en la tocata y fuga o en el recitativo y el aria. (González, 2021, p.48.)

Durante este periodo los instrumentos de cuerda alcanzan su máximo grado de perfección y estandarización, a expensas de mejoras mínimas en cuestión de tecnología de materiales en los últimos años.

Durante esta época surgieron los mejores constructores de instrumentos de cuerdas de todos los tiempos como Stradivarius, Amati y Guarneri; por eso la música llega a un gran virtuosismo o perfeccionamiento, y los instrumentos tienen que ser capaces de soportar la exigencia que la interpretación del músico le demande. (Ruiz, 2006, p. 13.)

La Influencia de la Imprenta en la Caligrafía Musical.

La importancia de la imprenta radica en el que el ser humano logró con ella pasar de los manuscritos ilustrados y con adornos precisos, a las primeras técnicas de impresión como la prensa de imprenta, la estandarización de la notación musical y los tipos móviles.

Johannes Gutenberg, nació en Alemania, Maguncia en el año 1400.

Presentó la prensa de imprenta que hacía uso de los tipos móviles, fundidos en unas matrices manuales y que eran fabricados con una aleación metálica de estaño, plomo y antimonio, idónea para dicha fundición y que permitía reutilizar la aleación para construir nuevas formas, entre los años 1436 y 1444. (Werner, 1998, p. 5.)

“Como una invención industrial que ayudaría a la producción de libros a gran escala, ya que había mucho dinero de por medio y personas dispuestas a sacarle provecho a la tecnología” (Velduque, 2011, p. 4.)

“La participación de Ottaviano Petrucci y John Rastell en las técnicas de imprenta de partituras dejó como resultado una notación musical uniforme, clara y consensuada para la impresión de partituras” (Luchetta, 2020, párr 5, 6.)

“En 1501, Ottaviano Petrucci publicó *Harmonice Musices Odhecaton A*, que contenía 96 piezas de música impresa” (Promocionmusical, s. f, párr. 25.)

El método de impresión de Petrucci produjo música limpia, legible y elegante, pero fue un proceso largo y difícil que requirió tres pasadas separadas a través de la imprenta, que comenzaba por la impresión de líneas, texto y notas, aunque los resultados eran muy limpios y elegantes. (Luchetta, 2020.)

El método que diseñó Pierre Attaignant fue distinto: “Líneas y notas formaban parte de un mismo carácter, por tanto solo era necesario una impresión. Este método fue uno de los más populares alrededor de 1520 y se mantuvo en uso durante 200 años” (Promocionmusical, s. f, párr. 29.)

Toda las modificaciones que sufrió la música durante la Edad Media, los aportes de Hucblado de Saint Amand, Guido De Arezzo, el desarrollo del Ars Nova, Renacimiento y los aportes de la música barroca como lo son la tonalidad, armonía, ornamentación, dinámicas, texturas, ritmo y la estandarización de los instrumentos, propiciaron la necesidad para buscar la manera de estandarizar la escritura musical, pero esto se consolidó con la aparición de la imprenta que logra darle claridad y estandarización a la tipografía musical.

Dichos estándares que se desarrollaron durante esos años permitieron y facilitaron la reproducción de partituras pero el tiempo que tomaba dicho proceso era algo lento y no todos

los músicos podían tener acceso a esta tecnología; sin embargo, la digitalización de los símbolos musicales a partir de los años 80, permitió a los músicos escribir mediante una computadora y reproducirlas partituras en tiempo real, lo cual permite ahorrar tiempo y crear partituras con un diseño perfecto y limpio ayudando a la lectura e interpretación de dicha composición o arreglo.

Programas de Edición y Producción Musical

El *software* en general y por definición es el conjunto de herramientas que permite controlar, crear y hacer múltiples trabajos de forma digital en una computadora y soportado por múltiples sistemas operativos. De igual forma debemos entender que dichos software están enfocados a determinados trabajos, por ejemplo. Procesadores de texto, editores de imagen, video, software de diseño asistido, software de edición de partituras y las estaciones de audio digital conocidos en inglés como *Digital Audio Workstation (DAW)*. Con estas dos últimas herramientas, con la primera podemos editar partituras haciendo uso de las fuentes musicales para crear, escuchar el audio aproximado de los que está representando en el trabajo e imprimir el resultado y con los software de audio podemos grabar sonidos que se pueden manipular y aplicar procesos de efectos con plugins externos o internos, que permite darle un carácter distinto al sonido. Asimismo este tipo de software tiene la capacidad de trabajar con instrumentos virtuales que son creados para poder reproducir los sonidos pregrabados de instrumentos específicos a través de las técnicas de sampleo y programación para ser usados con el protocolo MIDI.

En este trabajo se tomó en cuenta y se analizó la inclusión de las herramientas para dispositivos móviles; debido a su limitada capacidad para realizar un trabajo a nivel profesional no fueron incluidas, ya que este proyecto está enfocado en software con una capacidad de

trabajo más alta que los dispositivos móviles aún no pueden soportar durante largas sesiones de trabajo.

Principales Software de Edición de Partituras

Los software de notación musical son herramientas que se encargan de procesar las fuentes y tipografías musicales para la creación de partituras digitales, como si de un procesador de texto se tratase, permitiendo la creación de partituras complejas y de muy buena calidad, ahorrando tiempo y costos para los músicos o copistas.

En los siguiente párrafos podrá leer y conocer sobre la historia de los primeros software diseñados para esta tarea, tanto de licencia privada como de software libre.

Encore

Software de notación musical diseñado en los años 80 por la empresa Passport y luego comprada por la empresa GVOX. La primeras versiones estaban disponibles únicamente para Windows y las versión 5 es la más actualizada disponible para los sistemas operativos Windows 7 / 8 / 10 y Mac Os 10.6 en adelante, este software actualmente está descontinuado su desarrollo ya que la última actualización del Copyright es 2015, siendo un software de licencia privada. (STmidiMusic, s. f.)

Sibelius

Software creado en 1986. Desarrollado por Jonathan y Ben Finn. La primera versión del software es la versión 1.4 y solo fue soportada para los sistemas operativos Windows 98, Windows 2000, Windows XP, Mac OS 8 y OS 9. 1993 se funda Sibelius Software para vender Sibelius 7 en el Reino Unido, entre los primeros clientes se encuentran la editorial más grande de Europa, Music Sales, el compositor coral John Rutter y la Royal Academy of Music y 2006 Sibelius se convierte en parte de Digidesign, la división de audio de Avid Technology empresa estadounidense con sede en Burlington, Massachusetts. (Sibelius Software, s. f.)

La versión más reciente es 2021.9 publicada el 07 de agosto de 2021 y está soportada por los sistemas operativos Windows 10 y Mac OS, siendo un software de licencia privada. Actualmente Sibelius es uno de los software considerados como el estándar en la industria musical. (Avid Knowledge Base, 2021.)

Finale

Software creado en septiembre de 1988 en Bloomington, Minnesota, actualmente ubicado en Denver Colorado, desarrollado por Phil Farrand en las oficinas de Coda Music Software hoy en día llamada MakeMusic. La primera versión sería Finale 1.0 (Johnson, 2009, párr. 2, 5.) Posteriormente en 1992 se lanzó la versión Final 3.0, (Yoho, 2015, párr. 1.) En 1997 se lanzó la versión comercial 3.7, después de este lanzamiento las versiones eran identificadas por los años en las que serían lanzadas las versiones comerciales (Music Printing History, s. f, párr. 1.)

Finale está soportada por los sistemas operativos Microsoft Windows y Mac OS, siendo un software de licencia privada. Actualmente es un software muy popular en la industria de la música y la última versión que tiene soporte es Finale 27, publicado en junio de 2021, dejando de lado la identificación de las versiones por año. (MakeMusic, s. f.)

Lilypond

Software de notación musical diseñado y desarrollado en 1995, por Han-Wen Nienhuys y Jan Nieuwenhuizen. (Lilypond, 2002, párr. 1.) Sistema que se ejecuta sobre un archivo de texto que describe la música muy parecido a un lenguaje de programación y que permite editar las partituras en su totalidad gracias a su motor de edición. Se publicó bajo la licencia GNU GPL y la última versión estable es la 2.24.1, 12 de febrero de 2023, este Software está soportado por Microsoft Windows, Mac Os y Distribuciones GNU/Linux. (Lilypond, s. f, párr. 6.)

Guitar Pro

Software de notación musical diseñado en el año de 1997 por la empresa Arobas Music. enfocado en instrumentos de cuerdas como la guitarra, Banjo y Bajo. Su diseño está más enfocado a las tablaturas pero de igual manera a la notación tradicional. (Arobas Music. s. f.) La primera versión del software lanzada al mercado fue la 2.16 y solo era soportada por el sistema operativo Windows 95. (Internet Archive Wayback Machine, 1998.) La versión más actual es Guitar pro 7.6, publicado en abril de 2017 y está disponible para los sistemas operativos, Windows 7 SP1 / 8 / 10 y Mac OS 10.10 en adelante, siendo un software de licencia privada. (Arobas Music, s. f.)

MuseScore

Software de notación musical diseñado por el alemán Werner Schweer (Sourceforge, 2010, párr. 2.) Presentado en Octubre del 2002 la primera versión fue 0.0.0 publicado bajo la licencia pública general (GNU), la versión más actual MuseScore 3.6.2, publicado el 08 de febrero del 2021, disponible para sistemas operativos Microsoft Windows, Mac Os y Distribuciones GNU/Linux (musescore, s. f.) Este software actualmente cuenta con una plataforma en línea que permite el intercambio de partituras digitales llamada MuseScore.com, de igual forma tiene una presencia y aceptación moderada en las comunidades universitarias y de músicos profesionales que en algún momento lo han usado y otros lo utilizan como su editor principal, ya que al ser de licencia libre los usuarios pueden hacer uso de todas las herramientas que dispone este software.

Estaciones de Trabajo de Audio Digital (DAW)

Los *Digital Audio Workstation*, mejor conocidos como DAW, y traducido al español como “estaciones de trabajo de audio digital”, son una herramienta que permite la manipulación de muestras de audio digital mediante una computadora personal, estos software fueron

diseñados para simular las mesas analógicas de audio de los años 80, permitiendo a los ingenieros en audio poder mezclar y editar audio de forma más eficiente mediante estos sistemas y permitiendo ahorrar costos tanto en cantidad de equipos y espacio para un estudios de grabación hasta su evolución de hoy en día.

A continuación una breve historia de la evolución de los siguientes DAW más conocido y utilizado tanto para sistemas privados y free software en la industria de la producción musical.

Pro Tools

Pertenece a la marca Avid Technology, empresa estadounidense con sede en Burlington, Massachusetts. en su división llamada Digidesign, inició su primer intento con el desarrollo del software Sound Tools 1989 para plataforma Apple Macintosh, pero al pasar a la grabación multipista Pro Tools paso a sustituir dicho software en 1991 con una interfaz de cuatro canales. (Future Music, 2011, párr. 14, 20.)

La última versión es Pro Tools 2022.4.0, publicado el 27 de abril de 2022, es soportado por los sistemas operativos comerciales Microsoft Windows y Mac OS, y hace uso de licencia privada. (Avid Technology, Inc, s. f.)

El concepto de Pro Tools siempre se ha basado en la estación de trabajo de audio digital, la combinación de software y hardware creó un paquete completo a lo largo de la década de 1990 Digidesign (fusionándose con los gigantes audiovisuales estadounidenses Avid, ahora propietario de M-Audio, Sibelius y Euphonix) refinó continuamente tanto el front-end del software como las capacidades de E/S y DSP de la plataforma Pro Tools. (Gemtracks, 2021, párr. 16.)

FL Studio

Es un producto creado por Image-Line, con sede en Bélgica, se creó en 1998, con la versión 1.0 en ese mismo año, bajo el nombre de FruityLoops, se inició como un secuenciador

MIDI simple, pero cuatro años después cambiaría de nombre al actual y se daría soporte a multipista y MIDI, logrando así pasar a la categoría de DAW y siendo unos de los software más usados para la música electrónica. Teniendo una licencia privada y soporte único para las plataformas de Mac os y Windows, la última versión es la 20.9.2 publicada el 11 de marzo de 2022. (Image Line Software, 2022.)

Ableton Live

Se presentó en 2001, desarrollado por Bernt Roggendorf y Gerhard Behles en Berlín, Alemania. inicialmente el enfoque estaba más dirigido para trabajar con secuencias de audio pregrabadas, la versión Ableton Live 4 lanzada en 2004 fue la que más modificaciones presentó entre las cuales se incluyen: la interfaz MIDI y soporte para instrumentos virtuales, plugins VST y AU, Hoy en día es pionero en la presentaciones en vivo de muchos artistas musicales. (Future Music, 2011, párr 5, 9, 10.) Siendo un software muy usado en la actualidad para *Beatmaking* y la música electrónica. La versión más actual es Ableton live 11, publicado el 06 de febrero de 2018, es soportado por los sistemas operativos comerciales Microsoft Windows y Mac OS, y hace uso de licencia privada. (Ableton, s. f.)

Audacity

Desarrollado por Dominic Mazzoni y Roger Dannenberg en el año de 1999 en la Universidad Carnegie Mellon y que permite la captura de forma simple y rápida, con sus propios plugins. Fue lanzado como software de código abierto en SourceForge.net en mayo de 2000. Desde 2021, el desarrollo está a cargo de Muse Group. (Audacity, s. f.)

“Audacity se distribuye bajo los términos de la Licencia Pública General GNU (GPL), versión 2, está soportado por los principales sistemas operativos Microsoft Windows, Mac OS y distribuciones GNU/Linux. (License, 2021, párr. 1, 3). La última versión lanzada es Audacity 3.1 lanzada el 23 de diciembre de 2021. (Download, s. f.)

Ardour

El Software es el más avanzado en la comunidad open source y es tan bueno como otros DAW de licencia propietaria, desarrollado por Paul Davis en diciembre de 1999. Galisteo, Nuevo México. (Davis, 2019, párr. 1.)

Ardor está registrado bajo la Licencia Pública General GNU publicada v2 por la Free Software Foundation, soportado por los principales sistemas operativos Microsoft Windows, Mac OS y distribuciones GNU/Linux.

Cuenta con un equipo mundial que incluye músicos, programadores e ingenieros de grabación profesionales. Uno de sus principales contribuyentes es la marca de consolas Harrison, aportando importantes recursos financieros y de desarrollo al proyecto, SSL contrató a Paul para que trabajara a tiempo completo en Ardor durante el desarrollo de Ardor 2.0, al igual que SAE proporcionó fondos limitados que ayudaron a impulsar la compatibilidad con OS X nativo. Armand Klenk de SAE Munich fue responsable de rediseñar el modelo de edición básico de Ardour y proporcionar una gran cantidad de comentarios sobre el diseño de Ardour y la versión nativa de OS X, la marca Waves utilizó Ardour como punto de partida para su producto Tracks Live, y también ha proporcionado fondos en varias ocasiones para el desarrollo del proyecto. (Ardour, s. f, párr. 15 – 18.)

La última versión lanzada al mercado es Ardour 6.9, publicada el 12 de agosto de 2021. (Ardour, s. f.)

Reaper

Este software fue creado por Justin Frankel quien en 1990 creó Winamp el primer software en soportar la reproducción de archivos mp3. A finales del 2005 desarrolló Reaper para su uso propio pero permitió que más personas colaborarán con el desarrollo del software.

(Andersen, 2015). Creando en 2004 la empresa Cockos Incorporated en Rosendale, Nueva York. (Cockos Incorporated, s. f.)

El DAW está soportado por los siguiente sistemas operativos: Microsoft Windows, Mac OS y distribuciones GNU/Linux con la interfaz gráfica GTK + 3 y el motor de audio ALSA y PulseAudio. La última versión es Reaper 6.56, lanzada el 5 de marzo del 2022, hace uso de licencia privada para su comercialización y el uso de prueba gratis por 60 días que puede ser renovada aceptando los términos y condiciones del software. (Reaper, 2022.)

Studio One

En 2006 Jim Odom se asoció con KristalLabs Software Ltd, una empresa emergente fundada por Wolfgang Kundrus y Matthias Juwan, para comenzar a trabajar en Capture y Studio One. En 2008 se presentó el mezclador digital StudioLive 16.4.2 que hace uso de una licencia propietaria y bajo la marca de PreSonus empresa privada fundada en 1995, con sede en Baton Rouge, Luisiana, Estados Unidos. En 2009 se presenta el DAW con grabación y masterización integrada en la misma aplicación Studio One, inspirado en el mezclador digital StudioLive 16.4.2. En 2011 llega Studio One v2 con múltiples mejoras y corrección de errores, está soportado por los siguiente sistemas operativos; Microsoft Windows y Mac OS. (PreSonus, s. f.). La última versión es Estudio One 5.5.2, publicado el 5 de abril de 2022, compatible con Microsoft Windows y Mac OS. (Studio One, s. f.)

Con este breve resumen de la historia de los principales software de partituras y audio en la producción musical, se da a conocer el panorama de compatibilidad que tienen con las diversas plataformas de sistemas operativos y que tan versátiles son. De igual forma sabremos uno de los datos más importantes, qué tipo de licencia utilizan dichos software, por ejemplo el uso de licencia propietarias o licencia open source.

Sistemas Operativos

En este apartado encontraremos una breve historia de los principales *operating system* OS, traducido al español como “sistema operativo”, utilizados en todo el mundo y que permiten la interacción entre máquina y persona, haciendo de estos sistemas la función de traductor y dándole al usuario la capacidad de hacer múltiples tareas e instalar herramientas externas compatibles con dicha plataforma para aumentar sus prestaciones.

Mac OS X

En abril de 1976 se fundó la empresa Apple Inc. por Steve Jobs y Steve Wozniak, con sede en Cupertino, California. “La primera computadora personal que se presentó fue Apple I, abril de 1977 se presentó al público la Apple II convirtiéndose en el canon de lo que debía ser a partir de entonces una computadora personal” (Teresa, s. f, párr. 3, 5, 14.)

En 1996, Apple adquirió NeXT, con el fin de actualizar su sistema operativo, algo en lo que llevaba años trabajando pero que no había logrado completar. El sistema operativo diseñado por Jobs en NeXT fue el primer paso de su vuelta a Apple, y el primer paso hacia Mac OS. (Salces, 2021, párr. 5.)

“Mac OS X 1.0 Cheetah, presentado el 5 de enero de 2001 en la Macworld Expo de aquel año, llegó acompañado de términos como Aqua, Dock, Quartz y Darwin, siendo Darwin el núcleo de MacOS” (Salces, 2021, párr. 8.) “La última versión en la que la empresa trabaja actualmente es macOS 12.3.1 Monterey, publicado el 31 de marzo de 2022” (Apple Inc, 2022.) Debemos entender que las computadoras de Apple son un paquete único y de licencia privada, creado y distribuido por dicha empresa, las empresas externas que desean desarrollar para esta plataforma deben cumplir los estándares de calidad de Apple.

Windows

En 1975 se fundó la empresa Microsoft, en Albuquerque, Nuevo México, por Paul Allen y Bill Gates, responsables del actual sistema operativo Windows. (Microsoft Prensa, 2015, párr. 3.) “En 1981 IBM comienza a comercializar equipos con el sistema operativo de Microsoft “MS-DOS 1.0”, el software que será la base sobre la que se pueden ejecutar los programas del equipo” (Microsoft, s. f.)

El nombre que dieron a este nuevo sistema operativo es MS-DOS, “Microsoft Disk Operating System” (sistema operativo de disco de Microsoft). En noviembre de 1985, Microsoft lanza al mercado “Windows 1.0” y se presenta Microsoft Windows, una extensión del sistema operativo MS-DOS que proporciona un entorno gráfico. Ahora, en lugar de escribir comandos de MS-DOS, puedes desplazarte entre pantallas o “ventanas” señalando y haciendo clic con el ratón. (Nadella, 2015, párr. 6.)

“En julio del 2015 lanzan “Windows 10”, la versión más estable del sistema operativo y con una gran cantidad de software disponibles. Actualmente la última versión del sistema operativo es “Windows 11” presentado el 14 de junio de 2021” (Microsoft, s. f, párr. 58.) Windows cuenta con un soporte privado por dicha compañía y empresas externas que desarrollan software para múltiples propósitos.

GNU/Linux

¿GNU/Linux? Entender el significado de esta palabra es hablar de la historia de dos personas que cambiaron la industria de la tecnología y de cómo las limitaciones aplicadas por empresas privadas impedía el libre aprendizaje para poder controlar una computadora y obtener el mejor rendimiento de dicho hardware, es aquí donde el finlandés Linus Benedict Torvalds y el estadounidense Richard Matthew Stallman compartían la idea de que todo hardware creado por una empresa debería ir acompañado de un software y que este debería

ser de código abierto para que el usuario pudiera mejorar o corregir errores de dicho programa, pero muchas empresas optan por vender hardware con software básicos y limitados y el usuario tiene que hacer otro gasto para adquirir programas avanzados para poder trabajar.

“Linus Benedict Torvalds, inició el proyecto Linux en 1991 comenzó a trabajar en Linux como un sustituto del sistema operativo MINIX mientras estaba en la Universidad de Helsinki, en Finlandia” (Stephen, 2021, párr. 10, 11.) “Según se cuenta Linus Torvalds, al no poder costear la licencia de UNIX y cansado de las bajas prestaciones del sistema MINIX decidió construir su propia versión de UNIX, basándose en el código de MINIX, para computadoras AT-386” (Moreno, 2014, párr. 7.)

Se lanzó bajo la Licencia Pública General (GPL) de GNU, lo cual implica que todo el mundo puede ejecutar, estudiar, compartir y modificar el sistema de software. El código modificado se puede redistribuir e incluso vender, pero bajo la misma licencia. (Red Hat, 2018, párr. 3.)

Actualmente la versión del kernel es la 5.16.14, publicado el 11 de marzo de 2022, todas las versiones del Kernel Linux son públicas y de código abierto en la página The Linux Kernel Archives (The Linux Foundation, s. f.).

Richard Matthew Stallman inició el Movimiento del Software Libre en 1983. Es creador del proyecto GNU y presidente de la Free Software Foundation. Acuñó el concepto de copyleft y la licencia GNU GPL (Universidad de Chile, 2018.)

GNU es un sistema operativo de software libre, es decir, respeta la libertad de los usuarios. El sistema operativo GNU consiste en paquetes de GNU (programas publicados específicamente por el proyecto GNU) además de software libre publicado por terceras partes. El desarrollo de GNU ha permitido que se pueda utilizar un ordenador sin software que atropelle nuestra libertad. (Free Software Foundation, 2022,

párr. 1.)

El software libre es una cuestión de libertad, no de precio. Para entender el concepto, debe pensarse en «libre» como en «libertad de expresión», no como «cerveza gratis». Más precisamente, software libre significa que los usuarios de un programa tienen las cuatro libertades esenciales: La libertad de ejecutar el programa como lo desee, con cualquier propósito (libertad 0). La libertad de estudiar el funcionamiento del programa y modificarlo de modo que realice las tareas como usted desee (libertad 1). El acceso al código fuente es un prerequisite para esto. La libertad de redistribuir copias para ayudar a los demás (libertad 2). La libertad de distribuir copias de sus versiones modificadas a otras personas (libertad 3). Al hacerlo da a toda la comunidad la oportunidad de beneficiarse de sus cambios. El acceso al código fuente es un prerequisite para esto. Debido a la evolución de la tecnología y del uso de la red, estas libertades son ahora aún más importantes que en 1983. (Free Software Foundation, 2022, párr. 5.)

De los Instrumentos Físicos a los Controladores e Interfaces MIDI

Entender cómo la tecnología hizo evolucionar los instrumentos reales a controladores, instrumentos virtuales e interfaz MIDI, debemos remontarnos a los inicios de los primeros sintetizadores y de entender la necesidad de la humanidad de poder hacer uso de las computadoras y las automatizaciones que estas pueden lograr con una hardware o software dedicado para dichas tareas, dando inicio a este gran proyecto llamado *Musical Instrument Digital Interface*, mejor conocido como MIDI.

Nos remontan a 1860 cuando el científico alemán, Hermann Von Helmholtz, pionero en el uso de la electrónica para generar sonido, construyó varios equipos electromecánicos capaces de producir tonos puros con el uso de la electricidad. Más

tarde en 1897 en Iowa, Estados Unidos, Thaddeus Cahill construiría el primer sintetizador de sonido llamado Dynamophone o mejor conocido como Telharmonium, una máquina que era capaz de producir corrientes alternas a diferentes frecuencias. La evolución de estos sintetizadores se desarrolla de forma paralela a los avances en el campo de la electrónica y la necesidad de establecer mecanismos de control y comunicación entre diferentes dispositivos o entre las partes de un mismo equipo. (Moreno, 2013, párr. 1.)

Llegando a la década de los años 60 se marcó un nuevo comienzo para el diseño de los instrumentos electrónicos y capacidad para poder simular sonidos de mejor calidad y de forma simultánea, dos personajes que dan a luz sus primeros instrumentos de forma comercial, Bob Moog y Don Buchla. En esta época es cuando también se comienzan a utilizar de forma masiva los sintetizadores en la música en vivo y producciones musicales. Como consecuencia, dieron lugar a diferentes estilos como, New Wave, Techno Pop, Industrial y New Age. (The Bass Valley, 2021, párr. 1, 12.)

Robert Moog, Nueva York, U.S. Mayo de 1934 - 2005, Ingeniero en electrónica y doctor en ingeniería física. Logró en 1973 diseñar y vender sus primeros sintetizadores como el Moog system 55 y el Minimoog, siendo este último el primer sintetizador de tamaño portable y de los más vendidos y utilizados. (The Bass Valley, 2021, párr 7.)

Don Buchla, era en 1963 un estudiante de física en la universidad de Berkeley y que le permitirá tener contacto con la escena artística de San Francisco, California. Logrando desarrollar un proceso de síntesis que sirviera tanto para la composición como para la ejecución en directo, desarrolló un primer prototipo llamado Buchla's Music Box y tiempo después pasaría a ser su primer sintetizador llamado Buchla 100. (Blánquez, 2016, párr. 3.)

Teniendo las bases y la tecnología para poder sintetizar sonidos y que los músicos de esa época pudieran crear música, surgió la necesidad de poder comunicar los equipos entre sí y un grupo de ingenieros lograron otro hito de la revolución de los sintetizadores, llegando a crear el protocolo de conexión que hoy en día es llamado MIDI (Musical Instruments Digital Interface).

Tom Oberheim, nacido en Manhattan, Kansas, es un inventor, diseñador de instrumentos y pionero de la música electrónica estadounidense. Fundó Oberheim Electronics en 1970. La empresa con sede en Santa Mónica comenzó a producir pedales de efectos, secuenciadores digitales y el módulo de expansión del sintetizador (SEM), un módulo de sintetizador único diseñado originalmente para acompañar a ARP y Moogs. Sin embargo, Oberheim pronto integró el SEM en los primeros sintetizadores polifónicos disponibles comercialmente en el mundo: los revolucionarios Four Voice y Eight Voice. (Oberheim, s. f.)

El legado de innovación de Tom continuó a lo largo de la década de 1980, cuando Oberheim Electronics pasó de ser una pequeña operación a una empresa grande. Los dispositivos notables de esta época incluyen el DMX, una de las primeras cajas de ritmos digitales, el secuenciador polifónico DSX y la legendaria serie de sintetizadores OB-X. (Oberheim, s. f.)

Ikutaro Kakehashi, nació en Osaka, Japón en 1972, fue un ingeniero japonés y fundador de la empresa Roland siendo esta una de las grandes empresas fabricantes de software y hardware para la producción musical. Entre sus productos que cambiaron la manera de hacer música electrónica y que se consolidaron como verdaderos iconos de un cambio social y cultural a finales de la década de los 70 y se extendió por la de los 80 y que se llevó a cabo con instrumentos como la Roland TR-808, o la TR-909. (Gauzak, 2017, párr. 1, 2.)

Es el año de 1981 y los ingenieros Dave Smith, Ikutaro Kakehashi y Tom Oberheim, crearon un estándar de conexión entre instrumentos electrónicos y computadoras que fueran capaces de comunicarse con un protocolo para poder compartir información y crear música, facilitando a los músicos la posibilidad de poder componer música, crear sonidos y hacer secuencias en tiempo real. Ya que no había una compatibilidad entre los fabricantes y se restringía el uso de sintetizadores por parte de las personas, lo que en última instancia inhibiría el crecimiento de las ventas. Con la ayuda de las empresas Korg, Yamaha y Roland, estos ingenieros pudieron hacer de este protocolo una conexión universal entre los controladores, computadoras y software especializados. (The MIDI Association, s. f.)

Dave Smith, es el fundador de Sequential Circuits en 1977. Es conocido como la fuerza impulsora detrás de la especificación MIDI en 1981. Fue quien acuñó el acrónimo. En 1987 fue nombrado miembro de la Audio Engineering Society (AES) por su continuo trabajo en el área de la síntesis musical. Legado de innovación musical de Dave Smith, secuenciador analógico modelo 600 en 1974, fue el primer producto musical de Dave Smith, un secuenciador analógico de 16 pasos. Secuenciador digital modelo 800 en 1975, era un secuenciador digital capaz de almacenar 16 bancos de secuencias de 16 pasos para un total de 256 notas. Presentaba entrada tanto en tiempo real como por pasos. Programador modelo 700 en 1976, era un programador de sonido para Minimoog y ARP 2600. Podía almacenar un total de 64 programas en 8 bancos de 8 programas. Profeta-5, apareció en 1977, fue el primer sintetizador analógico polifónico totalmente programable del mundo. También se destacó por ser el primer instrumento musical con un microprocesador integrado. Estuvo en producción desde 1978 hasta 1984 y fue adoptado por músicos de todos los géneros. El Prophet-5 estableció el estándar para los polisintéticos programables. Incluso hoy en día, prácticamente todos

los sintetizadores polivinílicos pueden rastrear sus diseños hasta este innovador instrumento. (Sequential LLC, 2023.)

Hoy en día todo lo que rodea a este protocolo ya es de libre acceso para todos los interesados en poder crear sus propio sistemas, controlar instrumentos virtuales y poder componer música, ya que la única inversión para poder llevar a cabo esto es solo adquirir el hardware necesario y correrlo mediante una de las distribuciones de Linux. En este caso la más utilizada para la industria del audio y multimedia es Ubuntu Studio, esta plataforma basada en el kernel de Linux cuenta con todo un soporte para funcionar correctamente y todos sus manuales hoy en día se encuentran albergados en la red.

Capítulo II. Diferencias Entre Software Libre y Software Privado

Tipos de Licencias en la Industria del Software y la Música

La licencia de un servicio o producto es el acuerdo que el autor otorga a otra persona o empresa de poder utilizar lo antes mencionado. Esto aplica tanto en la música como en el software con un determinado propósito y durante un tiempo concreto, estas licencias son algo parecidas, ya que comparten ciertos rasgos, como la distribución, el tiempo de uso, los permisos de edición, reproducción o con fines comerciales, una pequeña diferencia en la música es que el autor de la obra firma un contrato con una distribuidora la cual se queda un porcentaje de las ganancias por distribución y ciertas cláusulas de exclusividad.

En estas dos industrias el consumo de estos servicios genera millones de dólares para las grandes empresas del entretenimiento que hacen uso del software y de la música para sus productos finales. Cualquier persona que se encuentre relacionada estas dos industrias puede generar grandes ganancias.

Una de las diferencias entre las dos industrias es la publicación del producto. Mientras que en la música el registro de una obra consiste en presentar el fonograma y el papel escrito que representa la obra para su respectivo registro y poder comercializar dicha obra, en la industria del software se vende un producto final, donde la empresa o desarrollador registra el programa y su código, pero solo vende el producto terminado donde el usuario no adquiere el derecho de modificar ni analizar el funcionamiento interno del software. La posesión de este documento es el único elemento legal que autoriza la tenencia y uso del software.

En los orígenes de la informática, el software era considerado un componente más de la máquina y los fabricantes de computadoras lo desarrollan y distribuyen, incluso poniendo a disposición de los usuarios el código fuente, casi sin restricciones en cuanto a su utilización o destino. Pero, para evitar inconvenientes por acciones

antimonopólicas, a partir de 1970 la firma IBM independizó la comercialización del software de la venta de sus equipos y comenzó a ejercer derechos legales sobre el mismo, iniciando así un cambio de modelo económico donde el software se desvincula del fabricante del equipo y se desarrollan aplicaciones que pueden correr sobre diversas plataformas. (Brocca y Casamiquela, 2005, p. 2.)

En este nuevo escenario, la mayoría de los diseñadores de software comienza a utilizar todos los medios técnicos y legales disponibles para proteger su desarrollo y mantener su posición en el mercado.

Licencia de Software Propietario

La mayoría de estas licencias están diseñadas y adaptadas por cada marca que crea y distribuye un software o plugin en la industria. Los rasgos más comunes son, los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuir (con o sin modificaciones), o que su código fuente no está disponible o el acceso a este se encuentra restringido. En el software propietario una persona física o jurídica (por nombrar algunos: compañía, corporación, fundación) posee los derechos de autor sobre un software negando o no otorgando, al mismo tiempo, los derechos de usar el programa con cualquier propósito; de estudiar cómo funciona el programa y adaptarlo a las propias necesidades (donde el acceso al código fuente es una condición previa); de distribuir copias; o de mejorar el programa y hacer públicas las mejoras (para esto el acceso al código fuente es un requisito previo)". (Ibarra, 2016.)

Las restricciones que pueden llegar a tener los términos y condiciones que aplican las licencias propietarias se encuentra en el caso de la marca Image Line quien es responsable del diseño del software FL Studio, en el siguiente párrafo se muestran un fragmento de una de sus políticas de uso:

“Acuerdo de licencia de usuario final del software image line.

7. Obligaciones del usuario. Párrafo tres. En el caso de que la versión del Software por parte del Usuario sea una versión de prueba o demo, el uso del Software por parte del Usuario se limita únicamente al uso privado. El Usuario tiene expresamente prohibido utilizar los proyectos, audio o video del Usuario creados con dicha versión de demostración del Software con fines comerciales, tanto con fines de lucro como sin fines de lucro. Sin perjuicio de las disposiciones anteriores, en caso de que el Usuario no cumpla con esta obligación, Image Line tendrá la propiedad total sobre estos proyectos y, por lo tanto, tendrá derecho a todos los ingresos generados por este uso comercial. Para que esto sea ejecutable, el Usuario entregará los controles de contabilidad del Usuario a Image Line por simple solicitud”. (Image Line Software, s. f. párr. 35.)

Estas restricciones afectan en otro aspecto, en el siguiente fragmento podemos leer una de las cláusulas en los Términos de licencia de usuario final del software Avid:

“Acuerdo de licencia para el software de avid.

3. Condiciones de la licencia.

3.10. Scorch. Si el software permite la creación de partituras musicales en formato Avid

Scorch, estas partituras no se podrán distribuir con fines comerciales”. (Avid, 2018, p. 219.)

Licencia Perpetua y Suscripción

Constantemente las marcas multinacionales buscan maneras más efectivas de poder generar más capital, mantener sus gastos operacionales, mantenimiento y los sueldos de los trabajadores, muchas de las empresas dedicadas a software de música iniciaron su modelos de negocios con la venta de licencias perpetuas.

El usuario final compra la licencia de uso de una versión particular de software, pagando en una sola exhibición el valor completo de dicha licencia. En la mayoría de los casos es necesario adquirir un contrato de soporte para tener acceso a correcciones o

parches para resolver problemas técnicos o solucionar vulnerabilidades de seguridad, ya que usualmente el periodo de garantía es corto, generalmente 90 días. En teoría, este tipo de licencias se pueden usar indefinidamente por no tener una vigencia, pero en la práctica la vida útil de una licencia de este tipo es corta y puede variar desde dos hasta cinco años, al final de la cual tenemos que adquirir una nueva licencia de uso para la última versión del software. (Marco, 2019, párr. 3.)

Ventajas: La inversión se realiza una sola vez y no se deben pagar montos posteriores (a menos que requiera actualizaciones o nuevos productos y/o servicios). No tiene un tiempo de caducidad determinado, la inversión realizada le servirá para muchos años.

Desventajas: El usuario no podrá acceder a nuevas versiones del software (por ejemplo actualizar de Finale 2010 a Finale v27); pero si así lo desea, tendrá que realizar una inversión adicional aunque este valor es inferior a que si comprara la nueva versión por primera vez. Dentro de este modelo, la pérdida de la llave HASP o el registro de software, significa la pérdida de la licencia. (Interpro, 2018, párr. 2 – 3.)

En el nuevo modelo por suscripción, el usuario paga a la marca una cuota mensual o anual por el uso de alguna de sus herramientas dependiendo el paquete que se eligió, le permite tener acceso a la última versión del mismo incluyendo correcciones a problemas o vulnerabilidades de seguridad y acceso a soporte técnico sin tener que contratar algún elemento adicional. (Marco, 2019.)

Ventajas: La inversión inicial es mucho menor que la que se debe realizar para adquirir una licencia perpetua. Está dividida en pagos anuales, mientras la suscripción esté vigente, tiene derecho a garantía técnica y soporte técnico. Si la suscripción está activa, tiene derecho a actualizar sin costo adicional a nuevas versiones del software. Si pierde la llave HASP o el registro de software, deberá cancelar únicamente el costo de

reposición. Desventajas: Para mantener activa la suscripción hay que realizar pagos consecutivos (normalmente anuales). (InterPro, 2018.)

Licencia de Dominio Público

Según la Free Software Foundation una licencia de dominio publico es está descrita como:

Es aquella que no tiene derechos de autor. Si el código fuente es de dominio público, se trata de un caso especial de software libre sin copyleft, lo que significa que algunas copias o versiones modificadas pueden no ser libres en absoluto. (Free Software Foundation, s. f. párr. 12.)

El software de dominio público puede ser libremente utilizado, copiado o alterado, porque legalmente nadie es dueño de los derechos para restringir su uso. Por dominio público se entiende la situación en que quedan las obras literarias, artísticas o científicas al expirar el plazo de protección del derecho de autor. Cada legislación nacional contempla un término de años contados desde la muerte del autor (post mortem auctoris), para que estos derechos expiren. En el caso de los software estos derechos no expiran por que son las empresas las que firman el acuerdo y no un propietario, cuando una empresa va a la quiebra estas son compradas por otra del mismo rubro quedándose con todas las licencias de la anterior empresa. (L & D, 2015, párr. 3 – 5.)

Licencia Permisiva

Este tipo de licencias son las principales licencias permisivas y comparten múltiples condiciones con licencias de software privado y software libre, una de las primeras licencias que aparecieron es la BSD (Berkeley Software Distribution) y la licencia MIT (Massachusetts Institute of Technology) a fines de la década de 1980. (Haff, 2019.)

No imponen prácticamente ninguna condición sobre quien recibe el software, y sin embargo le dan permiso de uso, redistribución y modificación. De hecho, estas licencias típicamente permiten la redistribución con licencia privativa de un software cuyo autor distribuye con licencia permisiva. (Barahona, 2012, párr. 1.)

Este tipo de licencias sólo se aplican al código fuente licenciado, y no hace intentos de interferir o aplicar condiciones sobre nada más, por lo que ahorran tiempo de definir lo que es una obra derivada, ya que no se aplica siquiera a ellas. (González, s. f.)

Con toda esta información damos por entendido que la mayoría del software que usa este tipo de licencias puede ser vendido y no hay obligaciones de incluir el código fuente, esto también suele pasar en la industria de la música cuando solo te venden los audios de una obra musical.

Licencias Restrictivas

Son aquellas licencias libres que forzan en sus cláusulas a la liberación de toda obra derivada de la original libre. Este aspecto las diferencia de las licencias permisivas, cuyas obras derivadas pueden ser privativas y garantizan que el acervo de conocimiento libre generará nuevo conocimiento también libre. Algunas licencias restrictivas principales, GPL y Mozilla, Las de uso general, FDL (equivalente a GPL) y Creative Commons Atribución Compartir Igual. (Alcántara, s. f.)

Licencia LGPL

Esta licencia solía llamarse Licencia Pública General de la Biblioteca GNU y apareció en junio de 1991. Esta es la primera versión publicada de la biblioteca GPL. Está numerado 2 porque va con la versión 2 de la GPL ordinaria. (Free Software Foundation, 1991.)

En su versión 2.1, publicada en febrero de 1999. Esta Licencia Pública General Reducida de GNU cuenta como sucesora de la Licencia Pública General de Biblioteca

GNU. Esta es la primera versión publicada de Lesser GPL. También cuenta como el sucesor de la Licencia Pública de la Biblioteca GNU, versión 2, por lo tanto el número de versión 2.1. (Free Software Foundation, 1999, párr. 5.)

El 29 de junio de 2007, aparece la versión 3. Esta licencia es un conjunto de permisos adicionales agregados a la versión 3 de la Licencia pública general de GNU (GPL), e incorpora los términos y condiciones de la versión 3 de la Licencia pública general de GNU, complementada con los permisos adicionales que se enumeran en dicha versión de la licencia. (Free Software Foundation, 2007, párr. 2.)

Este acuerdo se aplica a cualquier biblioteca de software, significa una colección de funciones de software y/o datos preparados para vincularse convenientemente con programas de aplicación (que usan algunas de esas funciones y datos) para formar ejecutables. de igual forma puedes redistribuirlo y/o modificarlo bajo los términos de GNU Library General Public, y a cualquier otra biblioteca cuyos autores decidan utilizarlo. Puedes usar se para bibliotecas personales. LGPL es usado habitualmente para licencias de componentes compartidos como por ejemplo librerías (.dll, .so, .jar, etc.).

Licencia Pública General (GNU)

“Esta licencia protege únicamente al software y su código. Apareció por primera vez en febrero de 1989” (Free Software Foundation, 1989). “En su versión 1 y la versión 2 apareció en Junio 1991” (Free Software Foundation. 2022). “La última modificación es la versión 3, 29 Junio 2007” (Free Software Foundation, 2007, párr. 3.)

“La licencia pretende garantizar que el usuario tenga la libertad para compartir, cambiar, mejorar y dar acceso libre al software y el código fuente” (Free Software Foundation. 1991.)

Para la protección de cada autor y la Fundación del Software Libre, asegura que todos entiendan que no hay garantía para estos software libres. Si el software es modificado

por otra persona y redistribuido, la Fundación de Software Libre quiere que sus destinatarios sepan que lo que tienen no es el original, por lo que cualquier problema introducido por otros no se refleja en el original (Free Software Foundation, 1989.)

“Para proteger sus derechos, necesitamos hacer restricciones que prohíban que nadie le niegue estos derechos o le pida que renuncie a los derechos” (Free Software Foundation, 1989.) “Esta Licencia Pública General se aplica a la mayor parte del software de la Free Software Foundation y a cualquier otro programa” (Free Software Foundation, 1991.)

Licencias Creative Commons






La organización fue fundada en 2001 por el profesor de la Universidad de Stanford Lawrence Lessig, junto con Hal Abelson y Eric Eldred. Las primeras licencias Creative Commons vieron la luz en 2002 y es una organización sin fines de lucro. (Riquelme, 2019, párr 4.)

Se creó con el propósito de poner a disposición de la comunidad internacional la información científica, de igual manera atendiendo las necesidades de proteger obras como música, imagen, video, video juegos entre otros tipos de obras comunes, sin las habituales restricciones por parte de derechos de autor. Basta que el autor o el titular de los derechos conceda una autorización para poner la información a disposición de los usuarios.

Las licencias Creative Commons brindan a todos sus usuarios, desde creadores individuales hasta grandes instituciones, una forma estandarizada de otorgar permiso público para usar los trabajos creativos bajo la ley de derechos de autor, tal y como se puede observar en las Tablas 1 y 2. En este siguiente apartado podemos ver los logos con los que se identifican las obras registradas bajo las licencias Creative Commons.

Tabla 1








Clasificación de los logos que conforman las distintas licencias Creative Commons.

Logo	Denominación	Descripción
	CC0	Puede copiar, modificar, distribuir y realizar el trabajo, incluso con fines comerciales, todo sin pedir permiso.
	BY	Se debe dar crédito a la creadora de la obra original.
	SA	Las adaptaciones deben ser compartidas en los mismos términos que el creador original establece para dicha obra.
	NC	Solo se permiten usos de forma no comercial de la obra a utilizar.
	ND	No se permiten derivados o adaptaciones de la obra.

Esta tabla permite conocer los distintos logos que usa Creative Commons para clasificar sus licencias y los permisos que le otorga a los usuarios que las usan en sus obras y colaboraciones.

Tabla 2

Tipos de licencia oficiales por Creative Commons.

Licencia	Permite	Siempre que
 Reconocimiento (BY)	Compartir (copias y redistribuir) y adaptar (remezclar, adaptar y desarrollen un obra a partir del material) e incluso con fines comercialmente.	* Se reconozca al creador de la obra original.
 Reconocimiento Compartir igual (BY-SA)	Compartir (copias y redistribuir) y adaptar (remezclar, adaptar y desarrollen un obra a partir del material) e incluso con fines comercialmente.	* Se reconozca al creador de la obra original. * Se licencie la nueva creación en su caso bajo Condiciones idénticas (BY-SA).
 Reconocimiento Sin obra derivada (BY-ND)	Compartir (copias y redistribuir) e incluso con fines comercialmente.	* Se reconozca al creador de la obra original. * No se distribuyan modificaciones de la obra original.
 Reconocimiento No comercial (BY-NC)	Compartir (copias y redistribuir) y adaptar (remezclar, transformar y crear a partir de la obra original).	* Reconocimiento al autor de la obra original. * No utilizare con propósitos comerciales.
 Reconocimiento No comercial Compartir igual (BY-NC-SA)	Compartir (copias y redistribuir), Adaptar (remezclar, transformar y crear a partir de la obra original).	* Reconocimiento al autor de la obra original. * No utilizare con propósitos comerciales. * Se licencia la nueva creación (en su caso) bajo Condiciones idénticas (BY-NC-SA).
 Reconocimiento No comercial Sin obra derivada (BY-NC-ND)	Compartir (copias y redistribuir) el material.	* Reconocimiento al autor de la obra original. * No utilizare con propósitos comerciales. * No se distribuyan modificaciones de la obra original.
 Sin derechos reservados (CC0)	Compartir (copias y redistribuir) y adaptar (remezclar, adaptar y desarrollen un obra a partir del material) e incluso con fines comercialmente.	* El autor original renunciar a los derechos de autor.

En esta tabla podemos obtener información más precisa sobre cómo funcionan las licencias para el usuario final que desea hacer uso para proteger sus obras originales u obras derivadas.

Con este análisis de los tipos de licencia podemos entender un poco del compromiso que las empresas, instituciones y personas tienen al vender un producto, para su uso particular

o comercial, al usar estas herramienta útiles para el trabajo y saber de antemano qué grupo de la sociedad es más responsable al otorgar los permisos o bloqueos de uso, para la utilización de su producto en beneficio del aprendizaje, si este es para la educación profesional donde muchos interesados suelen cambiar de carrera cuando están en este periodo de tiempo y genera un gasto que muchas veces es difícil de pagar y que algunas licencias no permiten su reventa a terceros usuarios, ya que sus licencias no permiten dichos movimientos de su producto.

Capítulo III. Presentación de los Software de Producción Musical

Iniciar con este capítulo y por la naturaleza de la industria el software privado resuelve de manera fácil y comercial la adquisición de herramientas digitales; pero gestionar los recursos necesarios requiere tiempo y esfuerzo que en la mayoría de los casos no son autorizados por la institución algo que también llega a suceder en instituciones privadas pero en menor medida, tener todas estas herramientas permiten a los usuarios no perder tiempo en las actividades, es por esto que tener software libre alternativos para su uso es una opción bastante viable.

Una de las formas más usadas y no éticas es optar por el uso de crack o parche que no ha sido desarrollado por el proveedor de un paquete de software, autor del software o programadores respaldados por alguna fundación de software y que se pueden buscar en internet y así poder tener acceso a todas las herramientas del software. De igual forma es muy importante para un usuario conocer en qué consiste una aplicación de este tipo para entender los riesgos a los cuales podría estar exponiéndose antes de decidirse por esta opción, ya que podría tener repercusiones de carácter legal, porque en México y otros países existen leyes que protegen los derechos de autor y es el reconocimiento que hacen en favor de todo creador de obras literarias, artísticas, tecnológicas, audiovisuales, fotográficas y otras más.

Es de saberse que en México el uso de estos programas no autorizados está prohibido no solo por la Ley Federal del Derecho de Autor, sino, también por el Código Penal Federal, cuyas infracciones son sancionadas, con multas y penas desde seis meses hasta 10 años de prisión. (Cruz, 2019.)

Otro problema generado por el mal uso de un software privado es la aplicación de cracks. Los más usados y que son generalmente del tipo troyano, siendo de los más peligrosos ya que ponemos en riesgo información personal que se guarda en las computadoras

personales. Ya que este tipo de técnicas lo que permite es liberar el software del bloqueo que comúnmente se aplica en cualquier videojuego, editor de imágenes, reproductores de música, soluciones de seguridad o cualquier otra aplicación con licencia privada, logrando que los usuarios lo instalen en su dispositivo, una vez infectado el sistema podría permitirle a los cibercriminales acciones malintencionadas como instalar otros códigos maliciosos, lanzar ataques DDoS en contra de otros equipos, registrar lo que teclea el usuario con herramientas conocidas como *keylogger*, captura de la cámara web, entre otras operaciones. (Gutiérrez, 2013, párr. 7 – 8.)

Es importante aclarar que los cracks pueden ser creados para cualquier tipo de plataforma, como los smartphones, computadoras de escritorio, tablets o laptops y los sistemas operativos más atacados son Windows, Android y macOS, no solo porque sean relativamente sencillo de vulnerar, sino porque son los más usado, por lo que hay más probabilidades de éxito para los cibercriminales.

MuseScore como Propuesta de Editor de Partituras

MuseScore, es de los software de edición de partituras más populares, debido a que es de código abierto con licencia GNU GPL. La última versión del código de MuseScore siempre se encuentra disponible en el repositorio de GitHub. Disponibilidad del manual del desarrollador para obtener instrucciones sobre cómo crear MuseScore para su sistema operativo.

MuseScore está desarrollado por una comunidad abierta la cual cuenta con desarrolladores, 33 colaboradores activos en el último año, Voluntarios que colaboran en el foro con una participación de hasta 3781 comentarios en 30 días y de igual forma con traductores 1254 traductores y la traducción del software en 69 idiomas. (MuseScore, 2022.)

Tal como se observa en la Figura 1, en la página web de descargas de MuseScore, se puede ver la variedad de plataformas principales en las que se puede ejecutar el software.

Figura 1

Información de la página oficial de descargas de MuseScore.

musescore [Subir](#) [Acceder](#)

La última versión estable es 3.6.2
Released [February 08, 2021](#) ([release notes](#))

Descargar para Windows (64-bit)
Windows 7 o más nuevo
[Paquete 32-bit](#)
[Microsoft Store](#)
[Versión portátil](#)
[Versiones anteriores](#)

Descargar para Mac
macOS 10.10 o más nuevo
[Versiones anteriores](#)

Descargar Appliance
Adecuado para todas las distribuciones de Linux

Linux ▾

AppImage (recommended)
Adecuado para todas las distribuciones de Linux
[Computadoras viejas \(32 bits\)](#)
[Nuevas computadoras \(64-bit\)](#)
[ARM-architectures](#)

Snappy
Adecuado para todas las distribuciones de Linux
[Instrucciones](#)
Run `sudo snap install musescore`

Flatpak
Adecuado para todas las distribuciones de Linux
[Descargar](#)

Ubuntu
[Paquete comunitario](#)
o busque "musescore" en el Centro de Software de Ubuntu.

Mint
[Paquete comunitario](#)
o busque "musescore" en Software Manager.

Arch Linux
[Paquete de distribución](#)
[Paquete comunitario](#)

Debian
El paquete "musescore" se puede instalar desde el repositorio principal usando su administrador de paquetes favorito o verificar la distribución "inestable" (sid).

Fedora
[Paquete de distribución](#)
or run `dnf install musescore`
or via the "Software" GUI package install tool.

Mageia
Descargue el paquete correcto de RPM utilizando el Centro de control de Mageia.

PCLinuxOS
Disponible en los repositorios PCLinuxOS.

Gentoo
Available in Portage. just run `"emerge musescore"`.

openSUSE
Install through YaST with Leap 15.1 +, or with the command `"sudo zypper in musescore"`. For Leap 15.0 [Community packages](#) are available.

***BSD**

OpenBSD
Install MuseScore from packages: `pkg_add musescore`
or ports: `audio/musescore`

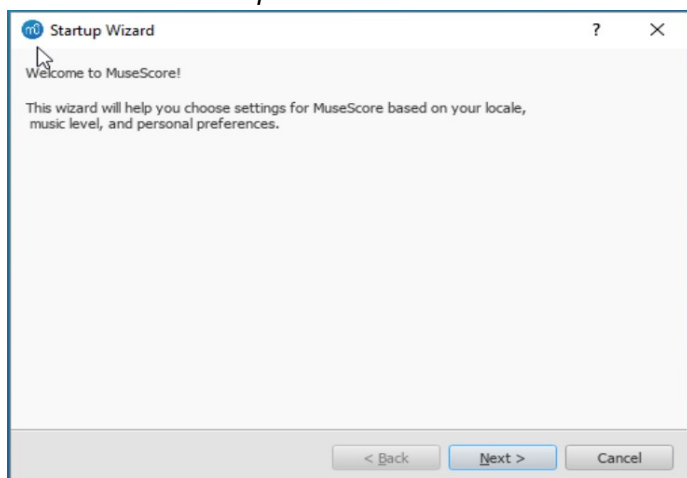
FreeBSD
Install MuseScore from the ports collection:
`audio/musescore`

DragonFly BSD
Install MuseScore from the ports collection:
`audio/musescore`

En la Figura 2, se puede observar el menú de configuración inicial. La primera vez que inicia MuseScore después de la instalación, se definen las preferencias básicas, como el idioma, tipo de teclado y el modo del espacio de trabajo, de igual forma puede activar el modo Tours, que permite conocer las funciones básicas de cómo manejar el software. estos pasos aplica para todas las plataformas en la que se puede utilizar.

Figura 2

Presentación de la pantalla de inicio de MuseScore.

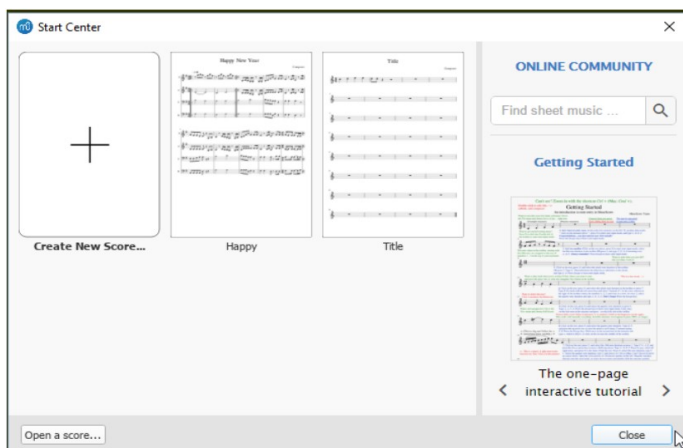


Presentación del Menú de preconfiguración cuando se inicia por primera vez el software.

En la figura 3 se muestra el centro de inicio. Esta ventana se muestra cuando se inicia MuseScore, permite crear nuevas partituras, proyectos, abre el asistente de partitura y permite ver los últimos proyectos en los que se ha trabajado.

Figura 3

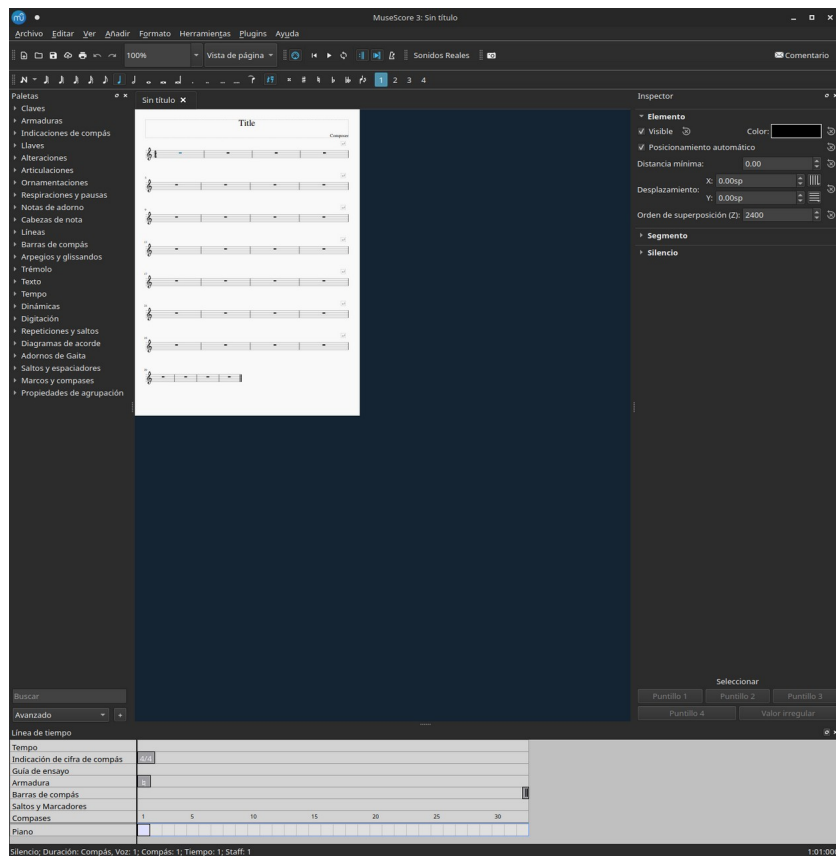
Presentación de la pantalla Centro de Inicio.



Presentación del área de trabajo de MuseScore en la cual se puede ver la barra del menú superior, donde se alojan todas las herramientas que se despliegan en pequeñas ventanas y también encontramos accesos directos como el poder reproducir, cambiar el tamaño de la vista, el tipo de nota a usar, alteraciones y los cambios de voces. En el menú lateral izquierdo se puede encontrar toda la simbología musical, desde las claves, barras de repetición, alteraciones, ornamentaciones, armaduras y mucho más. En el menú lateral derecho conocido como "inspector" encontramos las herramientas para editar cada detalle como color, distancia, tamaño de notas, grosor de las plicas. De igual forma se puede ver herramientas como el menú de reproducción y mezclador donde se pueden asignar las paquetería de sonidos, canales MIDI, paneo y el nivel de la ganancia de cada instrumento y por último en la parte inferior podemos ver la línea de tiempo de tiempo para poder desplazarnos con facilidad por la partitura. La mayoría de estos menús se pueden desacoplar de la ventana para poder mandarlos a una segunda pantalla o reubicar su posición en la pantalla de trabajo. Tal como se observa en la Figura 4, el entorno gráfico del software está preparado para crear una obra o algún trabajo específico.

Figura 4

Área principal de trabajo.

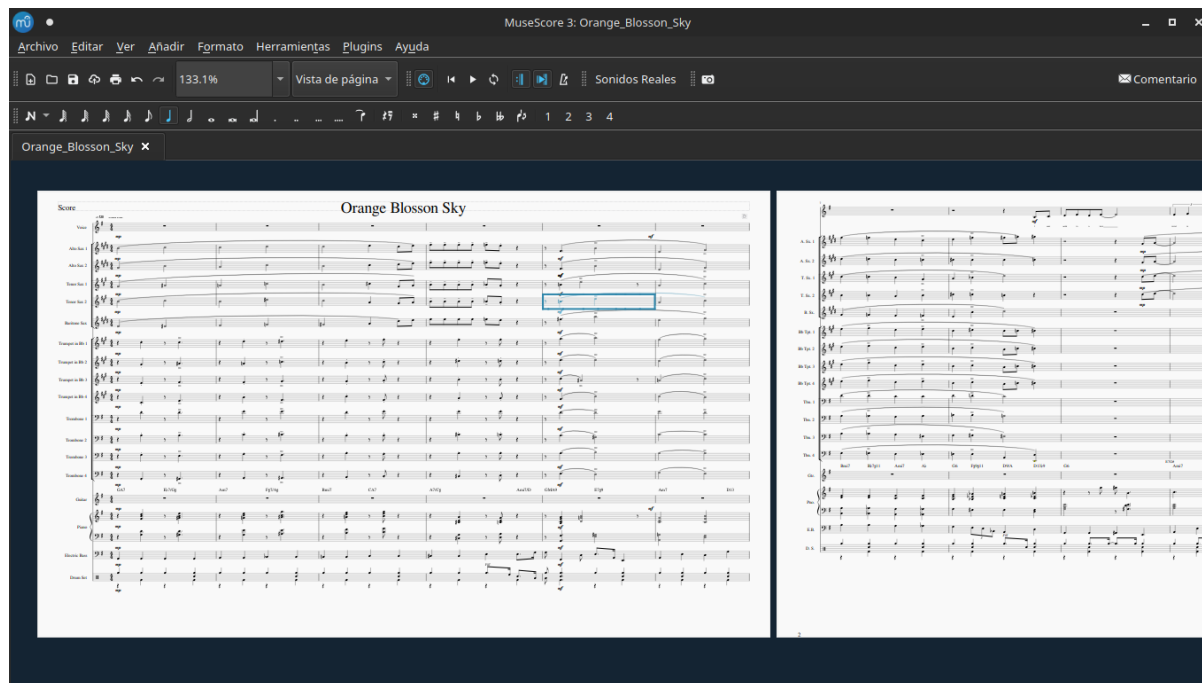


En el área de trabajo podemos encontrar todas las herramientas que son usadas para editar partituras.

En la Figura 5 se puede observar una transcripción para Big Band, creada en MuseScore, demostrando las capacidades del editor de partituras, estando al mismo nivel que sus contrapartes comerciales y en ocasiones superandolos. Tal como se observa en la Figura 5, una pequeña vista del entorno gráfico del software con un arreglo terminado, para después ser entregado en formato físico o digital.

Figura 5

Presentación del área de trabajo, con la transcripción de un arreglo para Big Band.



Se puede observar que las barras de herramientas laterales no están o fueron desacopladas para ser enviadas a un segundo monitor.

Como conclusión, podemos ver que MuseScore cuenta con las capacidades para poder crear obras en formato digital al alcance de todos los músicos sin importar el sistema operativo. De igual manera, soluciona la cuestión del uso de software libre contra software propietario en la universidad y todo lo que conlleva, permitiendo reducir los gastos y evitar cuestiones legales con las licencias.

Ardour: Propuesta de Estación de Trabajo de Audio Digital (DAW).

Ardour es un editor de audio profesional, desde 1999, aunque el estándar profesional por excelencia, desde hace muchos años, ha sido Pro Tools, una herramienta muy potente pero debido a su alto precio, sólo los profesionales de esta rama tiene la posibilidad de

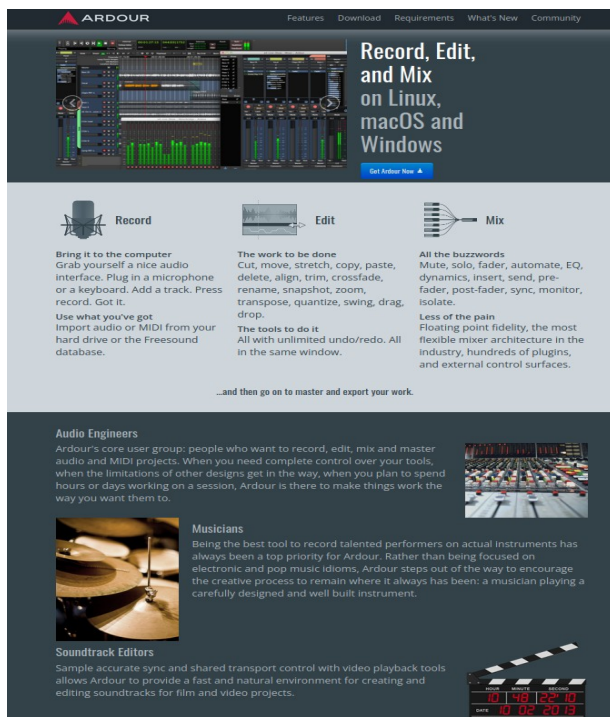
adquirirlo. En el extremo contrario está Audacity, un editor de audio básico y aunque no es el mejor, permite hacer cosas indispensables como, grabar en diferentes frecuencias de muestreo, cortar, copiar, mover audios y la aplicaciones de plugins. Y que se usan como alternativa libre cuando no se puede adquirir una DAW más profesional, de ahí su popularidad.

Entre los dos extremos, existe una alternativa que puede reunir lo mejor de ambos mundos como características profesionales, compatibilidad con diversas plataformas, traducido a múltiples idiomas y un bajo costo de adquisición. Ardour es un editor multipista más potente que Audacity y comparable a sus contrapartes privadas y con una interfaz y flujo de trabajo similar a los DAW más profesionales del mercado. Y algo a destacar es su precio de un dólar para su versión en Windows y Mac Os, mientras que para su versión en GNU/Linux, es de libre acceso e incluso viene preinstalado en distribuciones enfocadas a la producción de audio y video, como en el caso de Ubuntu Studio.

En la Figura 6, se muestra la página web principal y se puede observar un resumen de todas las capacidades que Ardour incorpora.

Figura 6

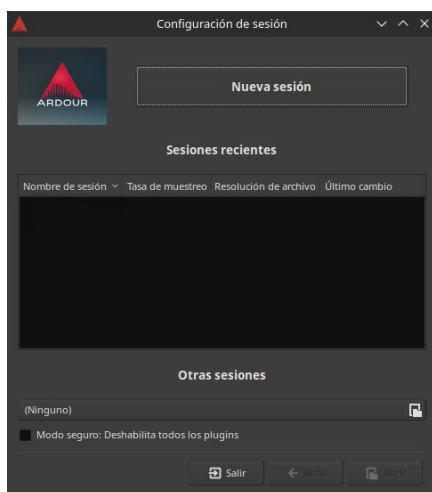
Presentación de la página web de Ardour.



En la figura 7, podemos ver la ventana de inicio de Ardour donde se pueden crear y alojarse todas las sesiones en la que se ha trabajado.

Figura 7

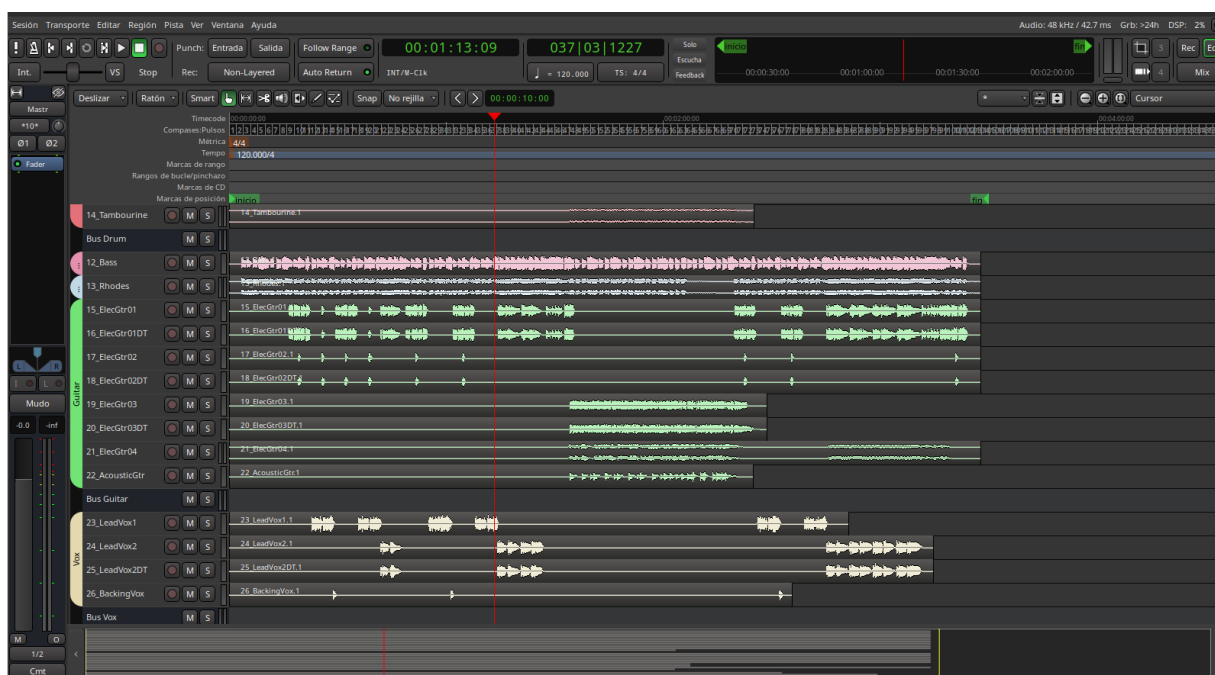
Ventana de inicio de Ardour.



En la figura 8, se puede observar el área de trabajo, donde igual se encuentra las herramientas utilizadas para poder manipular las pistas de audio y poder trabajar la sesión grabada.

Figura 8

Área de trabajo de Ardour.

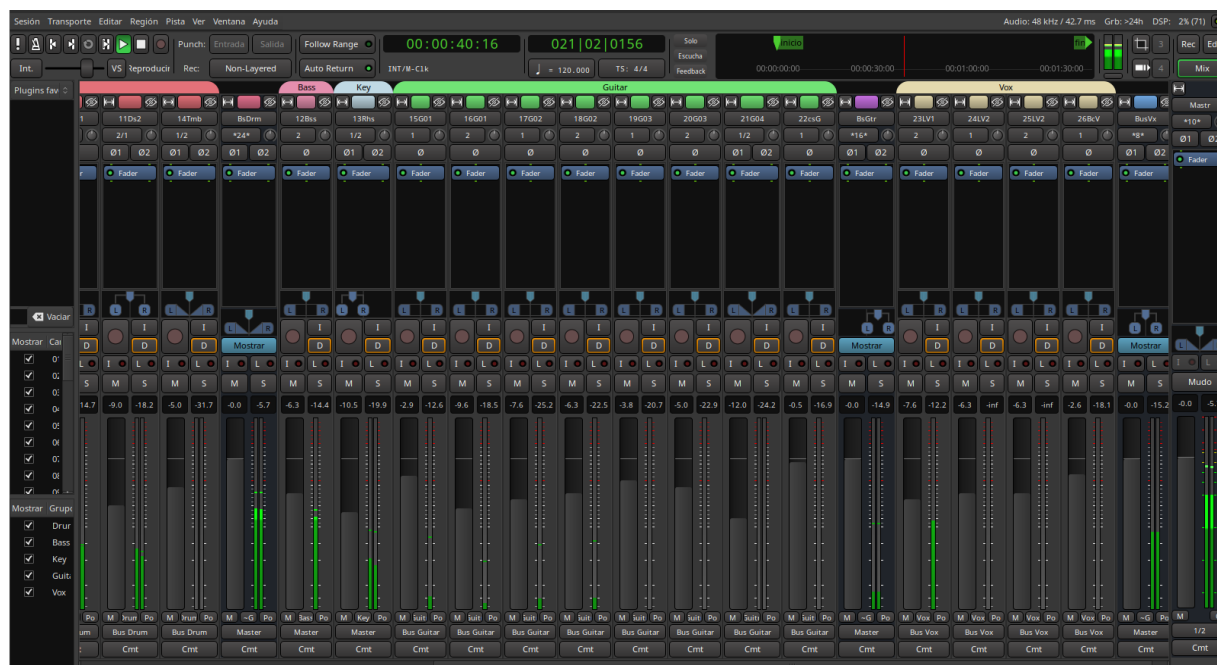


En la captura de pantalla podemos ver el trabajo realizado en una sesión de grabación y las herramientas que permiten trabajar con cada pista de forma independiente.

En la figura 9 se puede observar una de las áreas más importantes para la edición de audio que permite modificar el audio mediante la implementación de plugins y ruteos.

Figura 9

Área de mezcla y mucho más.



Como podemos observar, Ardour es un DAW capaz de sustituir a cualquier software privado de audio y permitiendo el acceso para todo aquel interesado en aprender producción y audio profesional, brindando todas las herramientas básicas para esta actividad.

Consideraciones Generales

Como se explicó en este capítulo y los anteriores, los software para la producción musical son un producto que permite y define la música, aunque no encasilla, porque se relaciona con preferencias y con estilos, que son parte de la identidad de los músicos y distintas áreas de la producción musical. Por tanto, se puede sostener que la industria del software interviene en una gran parte fundamental de la industria de la música.

En resumen, la mayoría de las personas escoge productos de las marcas más conocidas, sin saber por qué ese producto es de mayor calidad que otro y prefieren comprar

otro tipo de producto antes que confiar en marcas desconocidas. Teniendo en cuenta esto, se concluye que las marcas que tienen más publicidad son las más conocidas por las personas y por eso, son las más consumidas y más caras, sin antes probar el software libre y que se desconoce que de igual manera brinda las mismas herramientas, pero sin los costos exorbitantes que sus contrapartes de marcas privadas.

Capítulo IV. Implementación de Ubuntu Studio para uso Académico

¿Por qué la implementación de esta Plataforma?

Esta plataforma surgió por la necesidad de diversos artistas avanzados en el mundo del software, quienes se dieron a la tarea de desarrollar una plataforma versátil, que pudiese adaptarse a sus necesidades, y les permitiera conectar múltiples dispositivos sin las restricciones de las empresas con sistemas de licencia privada y que no perdieran capacidad de trabajo conforme se actualizaban el código fuente para mejorar la seguridad, rendimiento y de nuevas herramientas en la plataforma principal.

La base que permitió construir todo este ecosistema multimedia, inició tomando el diseño de una de las distribución GNU/Linux más famosa y fácil de usar conocida como Ubuntu, siendo respaldado por la empresa Canonical.

El nombre de esta distribución proviene de una antigua palabra africana que significa 'humanidad hacia los demás', y en ocasiones se describe como un recordatorio de 'soy lo que soy por lo que todos somos'. Llevando esta filosofía, significado y el espíritu de Ubuntu al mundo de las computadoras y el software. La distribución de Ubuntu representa lo mejor de una de las comunidades a nivel mundial del software libre que ha compartido su conocimiento con el mundo. (Canonical, s.f.)

Ubuntu Studio representa el esfuerzo de toda una comunidad que respalda el proyecto, siendo creado por voluntarios y dirigido a todos los niveles, desde principiantes hasta profesionales, y tiene como objetivo ser fácil de instalar y usar, así como proporcionar todas las herramientas necesarias para cualquier tipo de creación de contenido multimedia. El objetivo es ser un sistema operativo orientado y configurado para funcionar, listo para usar aplicaciones de audio, video e imagen, y que sea especialmente fácil de usar para los procesos complejos requeridos por los artistas a la hora de crear una obra artística, pudiendo ser un sistema casi

todo incluido. Esta es una de las tantas distribuciones GNU/Linux que son mantenidas por los donativos que hacen empresas y usuarios que confían en el Open Source, permitiendo tener así un sistemas dedicado y libre para el arte. (Ubuntu Studio, s.f.)

La Compatibilidad de las Interfaces de Audio con el Kernel de Linux

Para comenzar este tema, debemos tener en claro que hoy en día en la industria de la producción musical y otras profesiones el uso de una computadora es la herramienta por excelencia a utilizarse en todo momento, por eso la base de esto empieza por dos partes, el conjunto de hardware que constituye una computadora que sin un software que la controle por sí sola no cumple una función específica y el kernel o núcleo que se introduce con un sistema operativo, donde este es una parte fundamental del sistema operativo que se encarga de conceder el acceso al hardware de forma segura para todo el software que lo solicita, siendo este una parte muy pequeña e invisible para el usuario común, este se ejecuta en modo privilegiado con acceso especial para poder realizar las peticiones de acceso necesarios para las tareas en proceso que el usuario esté solicitando, decidiendo el orden de las peticiones recibidas según la prioridad e importancia de estas.

Actualmente existen tres kernel genéricos que son los más utilizados y que definen la computación moderna, siendo el kernel del Linux el más estable y utilizado. Cuenta con una variante de baja latencia, es de código abierto y las licencias que utilizan permiten al usuario común y avanzado poder modificarlo, si tienen los conocimientos necesarios para dicha tarea, permitiendo que los fallos de seguridad y errores sean corregidos casi de inmediato, de igual manera cuenta con voluntarios y empresas que dan soporte óptimo para plataformas doméstica o empresariales.

El segundo kernel más utilizado está en el ámbito de las computadoras domésticas, Windows NT, que es gestionado por la empresa Microsoft y que dispone de una licencia

propietaria. (Ranchal, 2014.) El tercer kernel está soportado por la empresa Apple que lleva por nombre XNU, haciendo uso de una licencia pública de Apple pero que es bastante restrictiva, cargado en diversos productos de la marca como los iMac, iPhone y iPad. (Velasco, 2017.) Para estos dos últimos solo las marcas y algunas empresas pueden tener acceso y modificar o sugerir mejoras, estos sistemas son atacados constantemente por los piratas informáticos, pero la empresas han podido lidiar con este problema, perjudicando al usuario final que no tiene control sobre este apartado.

Otro de los puntos importantes para el mundo del audio en una computadora es tener en cuenta el driver a utilizar para que la interfaz de audio interna o externa pueda operar en óptimas condiciones. En sus versiones de sistemas privados los driver tienen que ser específicos para cierto tipo de interfaces; algunas marcas suelen compartir los driver y en el caso de Ubuntu Studio y otras distribuciones GNU/Linux, hace uso del soporte ALSA.

Arquitectura avanzada de sonido de Linux (ALSA), siendo el módulo del Kernel encargado de comunicarse físicamente con el hardware de audio, proporciona funcionalidad de audio y MIDI a distribuciones GNU/Linux. Tiene las siguientes características significativas: Soporte eficiente de la mayoría de driver para todo tipo de interfaces de audio, desde tarjetas de sonido de consumo hasta interfaces de audio multicanal profesionales; controladores de sonido totalmente modularizados; biblioteca de espacio de usuario (alsa-lib) para simplificar la programación de aplicaciones y proporcionar una funcionalidad de nivel superior; compatibilidad con la antigua API Open Sound System (OSS), que proporciona compatibilidad binaria para la mayoría de los programas OSS. (ALSA project (Advanced Linux Sound Architecture), s.f.)

Teniendo en cuenta lo anterior pasamos a PulseAudio, estando en un nivel más alto siendo este el primer servidor de audio, situándose entre ALSA, el usuario y las aplicaciones

que reproducen o capturan audio. De las funciones más comunes y más apreciadas del PulseAudio es la capacidad para fijar y recordar los niveles de audio independiente para varios programas, la posibilidad de poder manejar cada interfaz y asignar tareas de manera independiente, pudiendo así aprovechar el potencial de cada interfaz. (López, 2019, párr. 4.)

Por último y el más importante es QjackCtl, también conocido como JACK, siendo el segundo servidor de Audio, cuya principal diferencia con PulseAudio, es que opera bajo un núcleo de baja latencia, que va instalado junto con el kernel genérico y que de igual forma es capaz de trabajar al mismo tiempo junto a PulseAudio, permitiendo tener todas la herramientas de este servidor, más las posibilidades que brinda QjackCtl, que hace uso del núcleo de baja latencia que le permite la captura de audio, monitores, envíos y señales MIDI en múltiples canales, ya sean virtuales o físicos, este último dependiendo del tipo de interfaz de audio a utilizar, todo en tiempo real dándole prioridad a todos estos trabajos sin verse afectado el desempeño de la computadora si es configurado correctamente, pudiendo llegar a un nivel de desempeño que dota al usuario con una herramienta integrada llamada Patchbay, lo que le permite gestionar todas las conexión libremente para cada programa y plugin cargado en el sistema. Todo esto es una pequeña parte con el que cuentan las distribuciones GNU/Linux enfocadas al audio.

Teniendo claro todo lo anterior podemos entender que el kernel más estable y versátil es el de Linux, ya que proporciona toda la documentación necesaria para poder operarlo, esto para las empresas se traduce en ahorro de recursos y tiempo, ya que la comunidad al tener acceso a la documentación de dicho hardware pueden diseñar y adaptar los driver y programas para ser integrados en las distribuciones GNU/Linux y que estos operen en óptimas condiciones dejando de lado la obsolescencia de los productos, culpa del mal soporte que las empresas destinan a este apartado.

Protocolo de Comunicación y Puertos E/S

Entendiendo todo lo anterior que tiene la importancia del rendimiento y compatibilidad de los driver con el kernel en un sistema operativo, pasamos a otro de los campos más importantes y que en muchas ocasiones no se le presta importancia y que ha estado en las computadoras desde sus inicios, ya que sin ellos no existiría la comunicación entre usuario y computadora, siendo estos los protocolos de comunicación o conexión, también conocidos como E/S (entradas y salidas) de la información y que para esto hay de muchos tipos y colores, pero nos fijamos en los más importantes que son USB, Firewire y Thunderbolt, que dotan al equipo de una capacidad de expansión gracias a estos protocolos y que permiten conectar múltiples periféricos.

Los puntos más importantes de estos periféricos son: el reloj que está encargado de marcar cada cuanto se realiza una transferencia de datos y los pines de datos encargados de transmitir información de un punto a otro. (Roca, 2022, párr. 5 y 6.)

FireWire

El puerto FireWire llamado así comercialmente por Apple, pero su nombre técnico es IEEE 1394, fue diseñado por Apple en 1986 en el Instituto de Ingeniería Eléctrica y Electrónica, conocido por sus siglas en inglés IEEE, siendo las iniciales que adopta su nombre técnico. Siendo uno de los primeros en ser lanzado la versión alfa se lanzó en 1995 con el nombre técnico mencionado. (Fernández, 2020, párr. 3.)

De este protocolo existieron varias versiones cada una con un factor de forma distinto, donde cada una de ellas mejora la velocidad de transferencia y estabilidad de conexión, actualmente no se utiliza, pero puede llegar a encontrarse en equipos antiguos.

Thunderbolt

Continuando con el protocolo Thunderbolt, fue creado por Intel en 2011 conocido igual por su nombre clave Light Peak y como sustituto del FireWire, con el fin de conectar diversos dispositivos a una computadora, proporcionando transferencias de datos, audio, vídeo en alta velocidad, superando a la competencia y la posibilidad de proporcionar energía a otros dispositivos. El primer lanzamiento salió con la ayuda de Apple en 2011, y ha estado casi como una exclusividad para algunos equipos de Apple. Este protocolo de conexión actualmente se encuentra en su versión 4, optimizando sus capacidades de conexión de diversos dispositivos y aumentando la velocidad de transferencia de datos, pudiendo sustituir a otros puertos. Esta última versión 4 garantiza una amplia compatibilidad con los sistemas operativos. (Alonso, 2020, párr. 7, 11.)

USB (Universal Serial Bus)

Llegando a uno de los protocolos más comerciales y conocidos del mercado, llamado USB (Universal Serial Bus), lanzado en 1996, cuando las PC comerciales estaban ingresando al mercado masivo a partir de los 1990, llegando a causar grandes problemas comerciales ya que aparecía un nuevo conector y dispositivo cada día, la entrega y administración de los controladores para todos estos dispositivos se convirtió en un problema muy rápido, ya que los usuarios de computadoras portátiles se quedaron rápidamente sin puertos seriales y paralelos. Esto no era tan importante para las PC de torre, pues los fabricantes de este tipo de computadoras incorporan una óptima cantidad de puertos seriales y puertos paralelos, pero no para las computadoras portátiles ya que la competencia de las marcas por vender portátiles se enfocó en el costo y tamaño de estos, afectando la cantidad de puertos, ya que eran componentes metálicos caros que ocupaban mucho espacio, y quedaba claro que había que

encontrar un reemplazo. (Logicbus, 2019, párr. 5 – 7.)

USB 1.0 apareció en 1996, logrando una velocidad máxima de 1.5 Mbit/s, dándole a las empresas un protocolo de conexión económico, estandarizado y fácil de emplear, frente a su rival, ya que FireWire contaba con mejores prestaciones pero no era tan económico de emplear. Posteriormente apareció la versión 1.1 incorporando notables mejoras como la implementación del *Plug and Play*, una velocidad de transferencia de hasta 12 Mbit/s.

Actualmente la última versión es USB 4.0, presentado en 2019, teniendo grandes mejoras a nivel de velocidad de transferencia de datos, llegando a rivalizar contra las prestaciones que ofrece la última versión de Thunderbolt. Una de las desventajas de esta versión es su implementación en los productos finales, ya que el usuario tarda en actualizar el hardware y la poca información que las marcas suelen brindar sobre sus productos, las versiones más populares actualmente y que aún están en uso son la 2.0 y la 3.1. (Fernández, 2021, párr. 4, 10.)

Formato SoundFont

Para poder introducirnos en la vanguardia de lo que fue la posibilidad de capturar el sonido de un ensamble de músicos o la captura de fuentes sonoras no musicales y la implementación de las computadoras para poder digitalizar el audio, esto fue la mezcla perfecta que le dio paso a las tecnologías para poder virtualizar y contener en un pequeño archivo o programa muestras de audio con alta fidelidad, lo que dotaría a las computadoras la posibilidad de poder simular instrumentos físicos de forma virtual o la simulación de procesadores de audio como lo son los ecualizadores, compresores, sintetizadores y muchos más, lo que también permitiría tener estaciones de trabajo mucho más compactas y más eficientes.

SoundFont

El formato SoundFont fue creado por las empresas, E-mu Systems hoy en día una subsidiaria de Creative Labs que fue fundada en Singapur en 1981.

Introduciendo dicho producto en la industria de la producción musical, este formato es capaz de contener una recopilación de audios, que se logra con la implementación de la técnica de Muestreo, la cual permite la captura de muestras de audio de cualquier instrumento u objeto que sea capaz de crear sonidos y que es muy utilizada en la industria del audio profesional (Martínez, 2014.) Esta tecnología hace uso del estándar MIDI, permitiendo gestionar hasta 16 canales y 128 muestras por canal en un solo archivo. Estos archivos comprimidos pueden ser cargados en plugins que sean capaces de hacer un mapeo MIDI para reproducir cada sonido como si del instrumento real se tratara. Un caso especial es cuando se trabaja con MuseScore que permite cargar este tipo de paquetería SoundFont para escuchar lo más similar posible el trabajo que estamos creando, o también puede ser el caso de conectar un controlador MIDI ya sea teclado o pads, para que este plugin con el archivo SoundFont cargado hagan el mapeado en dicho controlador para poder manipularlo.

Una de las características más importantes de esta tecnología es la capacidad de dotar al usuario la posibilidad de gestionar y crear sus propios archivos. Uno de los programas más utilizados en las distribuciones GNU/Linux es Polyphone, un potente editor de archivos SoundFont con la posibilidad de compartir y encontrar en los repositorios más paquetes de otros usuarios el programa utiliza la licencia GNU General Public License de código abierto. (Polyphone, 2020.)

Emulación de Instrumentos Virtuales, Formatos Comerciales y de Código Abierto

En el verano de 1996, una de las empresas más importantes patenta la primer tecnología que cambiará la industria del audio profesional, Steinberg pública bajo el nombre de

Virtual Studio Technology o mejor conocido como VST, un protocolo de expansión de plug-ins diseñado de forma específica para los software de audio, permitiendo que estos incorporen herramientas de procesamiento de audio o instrumentos virtuales junto con la intervención del protocolo MIDI. En lugar de usar hardware externo de procesamiento, los plug-ins VST hacen que los DAW aprovechen la potencia de procesamiento del ordenador para manipular el audio en tiempo real. De igual manera lanza el kit de desarrollo VST y abre la tecnología a otros fabricantes consiguiendo que el formato VST domine la industria de la producción musical. (Futuremusic, 2014, párr. 16 – 20.)

Actualmente esta tecnología evolucionó hasta el punto de que otras marcas han patentado sus propios protocolos de plug-ins con licencia privadas y de software libre, las versiones comerciales y de software libre en la industria actualmente son:

Virtual Studio Technology (VST)

Siendo el más popular en el mundo y diseñado por Steinberg, con una trayectoria de 25 años aproximadamente, y siendo el primero en ser libre, se implementó en casi todas los DAW existentes hoy en día, actualmente se encuentra en la versión 4.0. (FMCMstaff, 2014, párr. 16-20.)

El soporte de este formato de plug-ins está soportado por Linux y cuenta con toda la documentación para dicho trabajo, pero muchas marcas solo portean sus software para las plataformas comerciales, los usuarios más avanzados utilizan el soporte que depende del software Wine que es una capa de compatibilidad para los software nativos de Windows.

Audio Units (AU)

Es un protocolo diseñado y disponible solo para el ecosistema Apple, siendo este la competencia más directa de VST, teniendo muy poca diferencia para el usuario final. (O'Connor, 2020, párr. 11 y 12.)

AAX

Fue creado por Avid Technology en 2013 y sustituye a Real-Time AudioSuite RTAS y Time-division Multiplexing TDM, este formato actualizado a 64 bits logra lo que sus dos formatos anteriores pueden hacer al mismo tiempo. Este formato puede lidiar mejor con los proyectos de gran tamaño. Además, la gran ventaja de este formato es su capacidad de compartir sesiones entre sistemas DSP y Pro Tools nativos. (O'Connor, 2020, párr. 20 y 21.)

Formato de plug-ins para GNU/Linux

En este siguiente apartado se abordarán los formatos de plug-ins para todas las distribuciones GNU/Linux, que son un pilar importante para producción musical y que de igual manera podrán ayudar a los estudiantes e interesados en aprender todas las herramientas para lograr un trabajo de calidad.

LADSPA

Acrónimo de Linux Audio Developers Simple Plugin API. Se presentó el 2 de abril de 2000, como un estándar que permite conectar procesadores de audio y efectos de software, con un enfoque funcional, intuitivo y compatible con una amplia gama de software que permiten grabar y hacer síntesis de audio. Esta API está bajo la Licencia GNU Lesser General Public License. (Ladsp, s. f, párr. 13, 16.)

DSSI

Es un software de procesamiento, particularmente útil para complementos de síntesis de audio con interfaz de usuario, está específicamente bien documentado para su uso en aplicaciones de audio de Linux y se puede portear a otras plataformas. Se puede considerar como LADSPA para instrumentos, o algo comparable a VSTi. Este software es proporcionado bajo la licencia GNU Lesser General Public License. (Dssi, s. f, párr. 1 y 2.)

LV2

Es la combinación entre LADSPA y DSSI, cuya primera versión se lanzó en 2008, esto por la necesidad de un formato de plugin para plataformas GNU/Linux, que debería tener el poder de procesamiento de ondas de audio más la capacidad de generar sonido mediante eventos por MIDI. El proceso fue largo hasta que, finalmente, los mismos desarrolladores del formato LADSPA pusieron la primera piedra para construir el formato LV2. (Cespedes, 2014, párr. 10.)

SF2/3 y SFZ

Es un formato de tipo archivo y es usado para estructurar una colección de muestras para su interpretación, que puede ser creado con cualquier editor de texto o herramientas especializadas. El objetivo es proporcionar un formato libre, simple, minimalista y ampliable, con la capacidad de poder organizar, distribuir y usar muestras de audio de alta calidad. Los desarrolladores pueden crear, usar y distribuir los archivos de forma libre, ya sea para aplicaciones libres o comerciales. Este formato no es una fuente de sonido *Soundfonts*, ya que solo define el comportamiento de los instrumentos musicales y no incluye el contenido de las muestras. (Sfz format, s. f, párr. 2.)

Sin Limitaciones

Contar con un sistema operativo libre, dedicado y con todas las herramientas multimedia básicas y avanzadas disponibles, para poder crear una obra sin limitaciones. Todos este ecosistema cuenta con la capacidad de conectar periféricos externos, siempre y cuando su fabricante disponga o haga uso de los driver genéricos que la comunidad GNU/Linux crea para su correcto funcionamiento.

Los protocolos de conexión son importantes para la producción musical, ya que sin una buena velocidad de transferencia y estabilidad no se podría hacer la conversión AD/DA, que permite que las interfaces de audio tengan la mínima latencia posible para poder trabajar y que

va de la mano con el rendimiento del hardware, la optimización de los DAW y los plug-ins también conocidos como API, dotando al artista con la capacidad de capturar sus obras y poder trabajarlas de manera profesional cuando no se puede tener acceso a los estudios de grabación profesionales de las de empresas reconocidas, de igual forma el avance de la tecnología permite tener dispositivos más pequeños y de un costo accesible, cuando las herramientas son estandarizadas para el uso de estas.

Capítulo V. Pruebas de Rendimiento de Ubuntu Studio en Diferentes Gamas de Computadoras

Este capítulo presenta las evidencias del sistema operativo propuesto corriendo en diversos equipos de cómputo, tanto de marca comercial como ensamblados “a la medida”, utilizando como base la versión más reciente de nombre *Ubuntu Studio 22.04.1 LTS Jammy Jellyfish*. Esta versión puede ser descargada desde su página oficial a través del siguiente vínculo: <https://ubuntustudio.org/download/>.

Figura 10

Presentación de la página oficial de Ubuntu Studio.



En esta captura de pantalla se puede observar la página oficial y sus apartados en la cual podremos obtener toda la información necesaria y la página de descarga de la ISO.

Ubuntu Studio

El sistema operativo es de código abierto, el cual incluye una selección de programas multimedia preinstalados con licencia libre y configurados para obtener el máximo rendimiento para la edición y producción de audio, video e imagen profesional. Este sistema está construido a partir de un kernel de baja latencia, mejor conocido como “Kernel RT”, el cual permite la

sincronización en tiempo real para la conversión de señales de audio de AD/DA durante su grabación y monitorización.

Equipos y Hardware Externo de Prueba

En la siguiente lista se describen dos tipos de equipos, “gama baja” y “gama media”, con los que se puede ejecutar los software de audio y producción musical que incorpora Ubuntu Studio. Debido al contexto socioeconómico y a las pretensiones académicas de carácter principalmente didáctico e informativo del presente trabajo, los equipos de cómputo de “gama alta” no serán considerados para su evaluación, debido a su alto costo y prestaciones superiores necesarias sólo para uso profesional. Las especificaciones de cada equipo (gamas “baja” y “media”) nos permite comprender cuales son los requisitos para poder operar este sistema operativo y tener en cuenta las ventajas y desventajas de cada uno de estos equipos. Puntos a tener en cuenta al elegir los equipos: el procesador debe cumplir un mínimo de núcleos e hilos de subprocesos y tener en cuenta la antigüedad en la que salió al mercado. Para el apartado de memoria RAM no se mencionan marcas específicas ya que en el mercado solo existen pocos fabricantes de memorias confiables que son Samsung, Micron y SK Hynix, pero sí se recomienda un mínimo de 8 GB de memoria RAM repartido en dos módulos de memoria para activar el *Dual Channel* ya que es importante para evitar que los programas se congelen o tarden en responder. Para el disco principal donde estará el sistema operativo se recomienda el uso de SSD (*Solid State Drive*), ya que es importante la velocidad de lectura, escritura y el ancho de banda, que permite que los software puedan operar de forma fluida cuando se trabaja en baja latencia, algo que su contraparte de discos mecánico carecen, se recomiendan tres marcas y modelos, ya que solo hay pocos fabricantes de confianza que son Intel, Samsung y Micron.

Si se opta por un portátil como lo son las laptop es normal encontrar estos equipos con las mismas especificaciones que una computadora de escritorio; pero cuya principal diferencia radica en que la potencia del equipo es de un 15% hasta 30% menor que su contraparte, ya que estos equipo tienen que reducir costos y potencia para incorporar todos los elementos de una computadora de escritorio en un espacio muy reducido.

Equipos de Gama Baja

Los equipo de gama baja se destacan por ser equipos de bajo presupuesto y accesibles para la mayoría de personas; uno de los problemas que presentan estos equipo es su baja potencia de procesamiento lo que limita la posibilidad de mantener múltiples programas en ejecución y que su tiempo de vida útil sin perder rendimiento es corto, para el caso de la producción musical y audio en Ubuntu Studio estos equipo tienen los recursos suficientes para mantener sesiones de trabajo de perfil bajo. En la tabla 3 se presentan dos tipos de equipo de gama baja.

Tabla 3

Computadora de gama baja.

Computadora de gama baja.			
	Marca	Modelo	Especificaciones
Procesador	Intel	Comet Lake Core i3-10320	3,8 GHz, 4 núcleos y 8 hilos.
	AMD	Ryzen 3 3200G	3,6 GHz, 4 núcleos y 4 hilos.
Memoria RAM	Adata	AD4U2666W4G19-S	DDR4, 4GB, Velocidad de 2666 MHz
	Kingston	KF426C16BB/4	DDR4, 4GB, Velocidad de 2666 MHz
	HyperX	HX426C16FB3/4	DDR4, 4GB, Velocidad de 2666 MHz
SSD para el Sistema operativo	Adata	SU630 ASU630SS-240GQ-R	Interfaz SATA 6Gb/s Velocidad de lectura 520MB/s Velocidad de escritura 450MB/s
	Kingston	SA400S37/240G	Interfaz SATA 6Gb/s Velocidad de lectura 500MB/s Velocidad de escritura 350MB/s
	Western Digital	WDS240G3G0A	Rendimiento de lectura secuencial 545MB/s

Equipo de Gama Media

Este tipo de equipos son los más utilizados porque su rendimiento permite ejecutar múltiples programas de forma simultánea, hasta sesiones de trabajo complejas sin la preocupación de que el equipo muestre problemas de rendimiento. Algo a tener en cuenta en estos equipos es que son de un presupuesto relativamente alto pero que se compensa por su rendimiento y su tiempo de vida útil para el trabajo. En la tabla 4 se presentan dos tipos de equipo de gama media.

Tabla 4

Computadora de gama media.

Computadora de gama media.			
	Marca	Modelo	Especificaciones
Procesador	Intel	Comet Lake Core i5-10400	2,9 GHz, 6 núcleos y 12 hilos.
	AMD	Ryzen 5 5600	3,9 GHz, 6 núcleos y 12 hilos.
Memoria RAM	Adata	AD4U26668G19-SGN	DDR4, 4GB, Velocidad de 2666 MHz
	Kingston	KF426C16BB/8	DDR4, 4GB, Velocidad de 2666 MHz
	HyperX	HX432C16FB3/8	DDR4, 4GB, Velocidad de 3200 MHz
SSD para el Sistema operativo	Adata	SU630 ASU630SS-240GQ-R	Interfaz SATA 6Gb/s Velocidad de lectura 520MB/s Velocidad de escritura 450MB/s
	Kingston	SA400S37/240G	Interfaz SATA 6Gb/s Velocidad de lectura 500MB/s Velocidad de escritura 350MB/s
	Western Digital	WDS240G3G0A	Rendimiento de lectura secuencial 545MB/s

La información que se recopila para estos equipos cambia cada año, por la demanda de potencia que los software requieren para seguir operando conforme se actualizan a versiones nuevas. En el caso de las distribuciones GNU/Linux la demanda de potencia de procesamiento no es tan pronunciada pero no exenta.

Interfaces de Audio Externas

En esta sección se exponen las características de cada interfaz de audio probada. En su mayoría usan el protocolo de conexión USB 2.0, ya que es el más estandarizado, utilizado en el mercado, de los más antiguos, con soporte y la compatibilidad con sus versiones más recientes como el USB 3.1 , las frecuencias de muestreo más utilizadas son 44100 Hz y 48000 Hz con profundidad de 16 bit a 24 bits. Esto permite que los archivos de las sesiones a trabajar no ocupen tanto espacio y recursos de procesamiento de los equipos utilizados.

Tabla 5

Interfaces de audio probadas.

Interfaces de audio.				
Marca	Modelo	Protocolo de conexión	Frecuencia de Muestreo	Profundidad de bits
Steinberg	UR22C	USB C 3.0	44,1 kHz / 192 kHz	32 bit
Presonus	StudioLive 16.0.2	USB B 2.0	44,1 kHz / 48,0 kHz	24 bit
Behringer	UMC202HD	USB B 2.0	44,1 kHz / 192 kHz	24 bit
Focusrite	Scarlett 18i8 1ra gen.	USB B 2.0	44,1 kHz / 192 kHz	24 bit

La mayoría de las interfaces de audio tanto externas como internas para una computadora pueden superar estas especificaciones, pero es importante tener en cuenta estos parámetros para poder realizar capturas y la reproducción del sonido de forma óptima.

Uso del Hardware y Software Propuesto para la Producción Musical

La lista de software que a continuación se presenta, es el contenido que incorpora el sistema operativo; de igual forma, se pueden instalar otros software que sean compatible o buscando en el centro de software que incorporan las distribuciones GNU/Linux y como una

alternativa más haciendo búsquedas por la web y foros especializados se pueden encontrar otro tipos de software alternativos o que el sistema operativo no trae preinstalado.

Cuando el objetivo es crear una obra musical en un procesador de partituras digitales y posteriormente grabar, mezclar y masterizar, Ubuntu Studio dispone de los software profesionales para dichas tareas. También cuenta con la capacidad de procesar audio en vivo, codificar audio y video o hacer videollamada o *stream* con calidad profesional. Todo esto también se suele encontrar en otras distribuciones GNU/Linux pero no se llegan a configurar de forma óptima durante la instalación ya que en ocasiones se usan paquetes que los instaladores no incorporan. Un caso muy habitual es cuando se quiere correr el DAW de Ardour pero faltando la instalación del kernel de baja latencia; el software es funcional, pero ciertas funciones del DAW se ven limitadas.

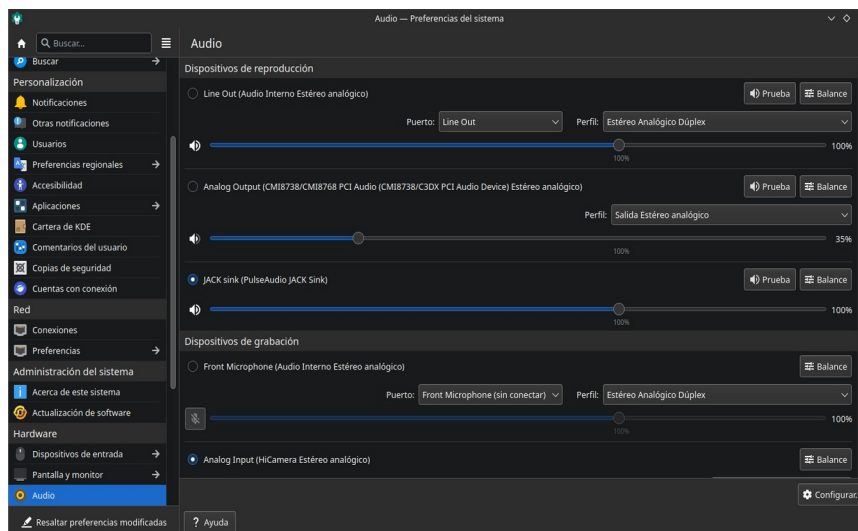
Apartado Gráfico de PulseAudio y ALSA

Recordando lo que se investigó en un capítulo anterior sobre los driver y las interfaces que pueden llegar a controlar los distintos Hardware internos como externos que componen una computadora, es momento de poder ver las distintas formas gráficas para gestionar los dispositivos en Ubuntu Studio.

Configuración del audio, dentro de las preferencias del sistema, es un apartado donde se puede modificar gran parte del comportamiento del equipo, la parte más importante para la producción musical es el apartado de audio, ya que aquí es donde aparecen todos los dispositivos de audio cuando son compatibles, si el hardware no tiene soporte este no aparecerá y damos por entendido que no funcionara, una última alternativa sería intentar buscar los driver e instalarlo de forma manual.

Figura 11

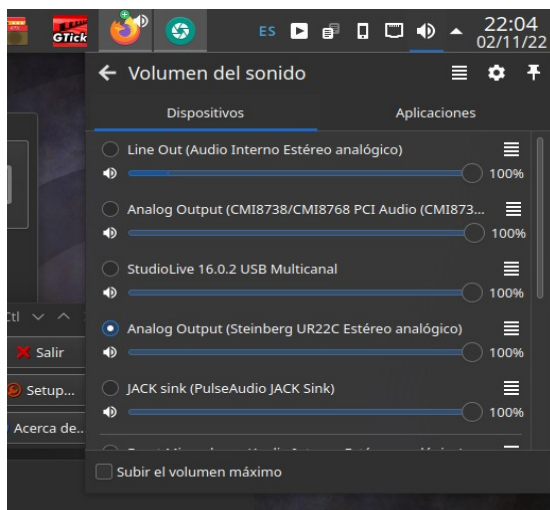
Preferencias del sistema, Audio.



En el apartado de audio se configura y gestiona tanto el audio como el hardware interno como externo los distintos perfiles, puertos de cada dispositivo y la capacidad de redireccionar el audio de las aplicaciones a las distintas placas de audio. Toda esta gestión se puede realizar desde la barra de tareas del escritorio, permitiendo hacer ajustes más rápidos y precisos, pudiendo sacar provecho de todas las salidas de audio del equipo.

Figura 12

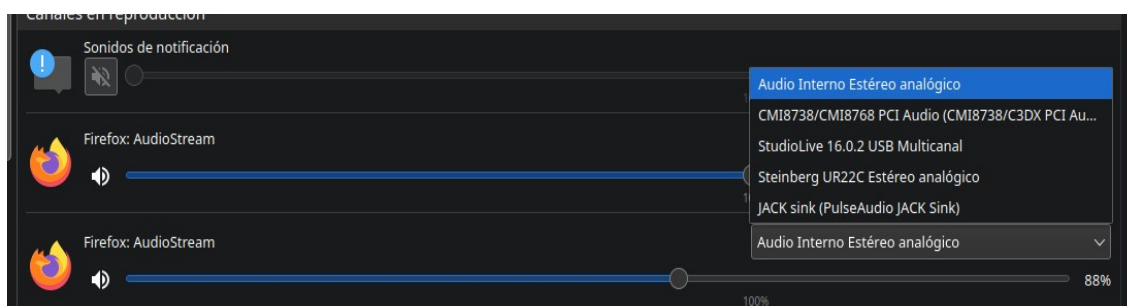
Gestión del hardware y los perfiles en la barra de tareas del escritorio.



Al trabajar con QjackCtl debemos tener en cuenta que al levantar el servidor de audio este esclaviza la interfaz de audio con la que vamos a trabajar por ende al conectar una aplicación la conexión se tiene que mandar a *Jack sink* el encargado de gestionar esta conexión.

Figura 13

Gestión y configuración del audio en cada aplicación.



Cada interfaz de audio y sus perfiles no solo se queda en el ajuste del Hardware, también para cada aplicación que estemos usando, permitiendo gestionar a donde se manden esas señales de audio de forma inmediata. En este apartado se puede gestionar qué Hardware

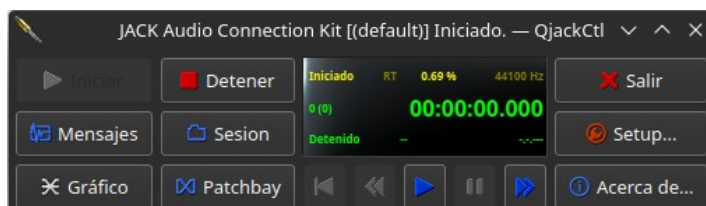
asignaremos para cada programa que esté en ejecución, de igual forma este apartado está disponible en la barra de tareas de Ubuntu Studio tal como aparece en la figura 12 pero en la segunda pestaña.

QjackCtl

También conocidos como Jack, es la parte del sistema operativo encargada de administrar el servidor de audio y MIDI con la capacidad de baja latencia y lograr así un desempeño profesional. Permite que todas las aplicaciones compatibles con Jack se conecten entre sí, además de permitir también el uso de varios dispositivos de audio a la vez en conjunto con PulseAudio y ALSA lo que le permite reconocer dispositivos de audio *Plug and Play*.

Figura 14

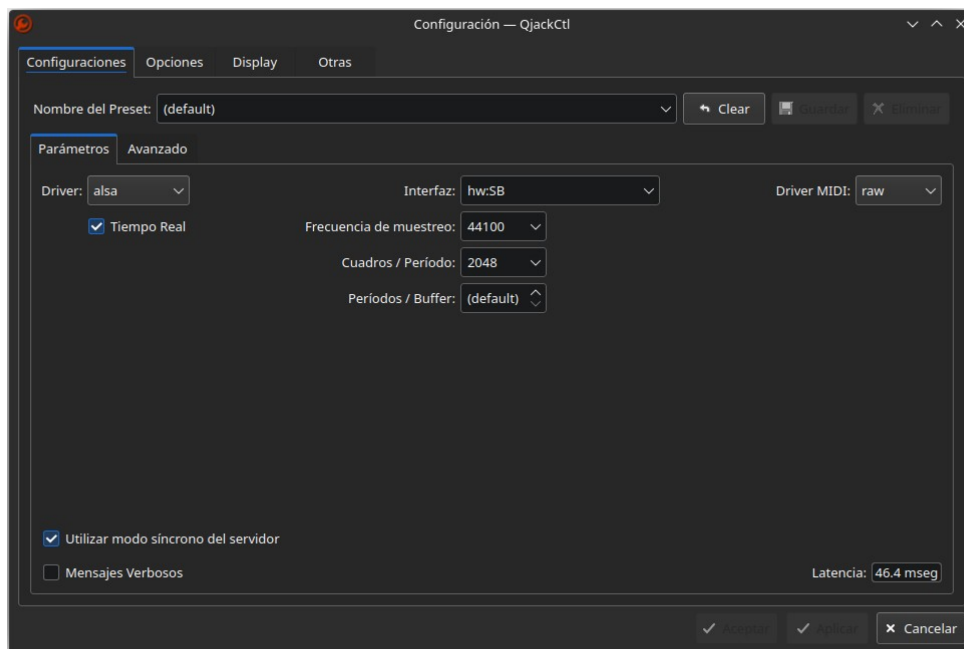
Presentación de QjackCtl en funcionamiento.



Podemos ver que a simple vista es software básico que su única funcionalidad es levantar el servicio de baja latencia del kernel pero que explorando sus funciones puede gestionar la entradas y salidas de audio y MIDI de forma libre.

Figura 15

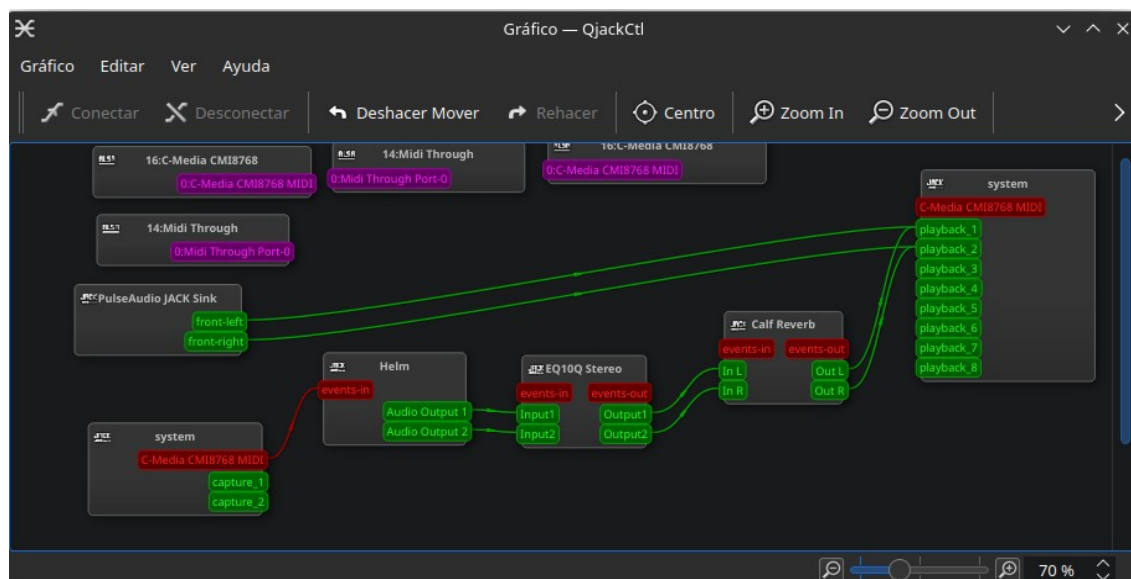
Configuración de QjackCtl.



En este apartado podemos seleccionar la interfaz de audio que se usará para trabajar con el núcleo de baja latencia, nos permite cambiar la frecuencia de muestreo y el tamaño del Buffer para bajar la latencia.

Figura 16

Patchbay gráfico.



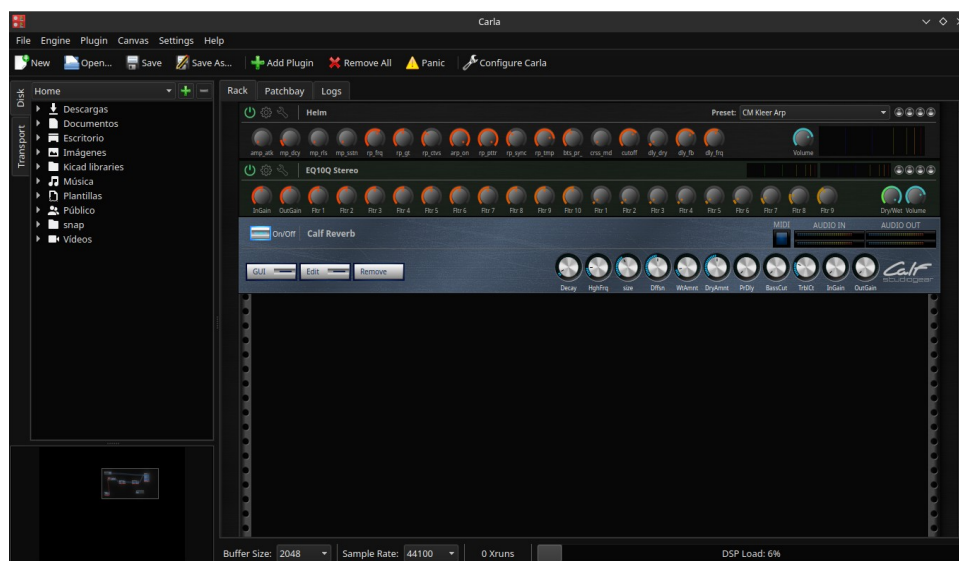
El patchbay gráfico de QjackCtl permite *gestionar los enrutamientos de la señales de audio y MIDI*, pero no permite cargar plugins o instrumentos virtuales, para poder lograr esto tenemos que hacer uso del software Carla que a continuación se mencionara.

Carla

Carla es un complemento de audio que al usarse junto a QjackCtl permite ver todas las conexiones específicas a nivel de *room control*, tales como: controlar plugins de manera externa al DAW, ruteo de parámetros a través de MIDI CC, control remoto sobre OSC, señales de audio y conexiones a otras aplicaciones que incorporen este modo de trabajo. Este software permite, además, ver todas las conexiones que hace el DAW en una sesión de trabajo activa.

Figura 17

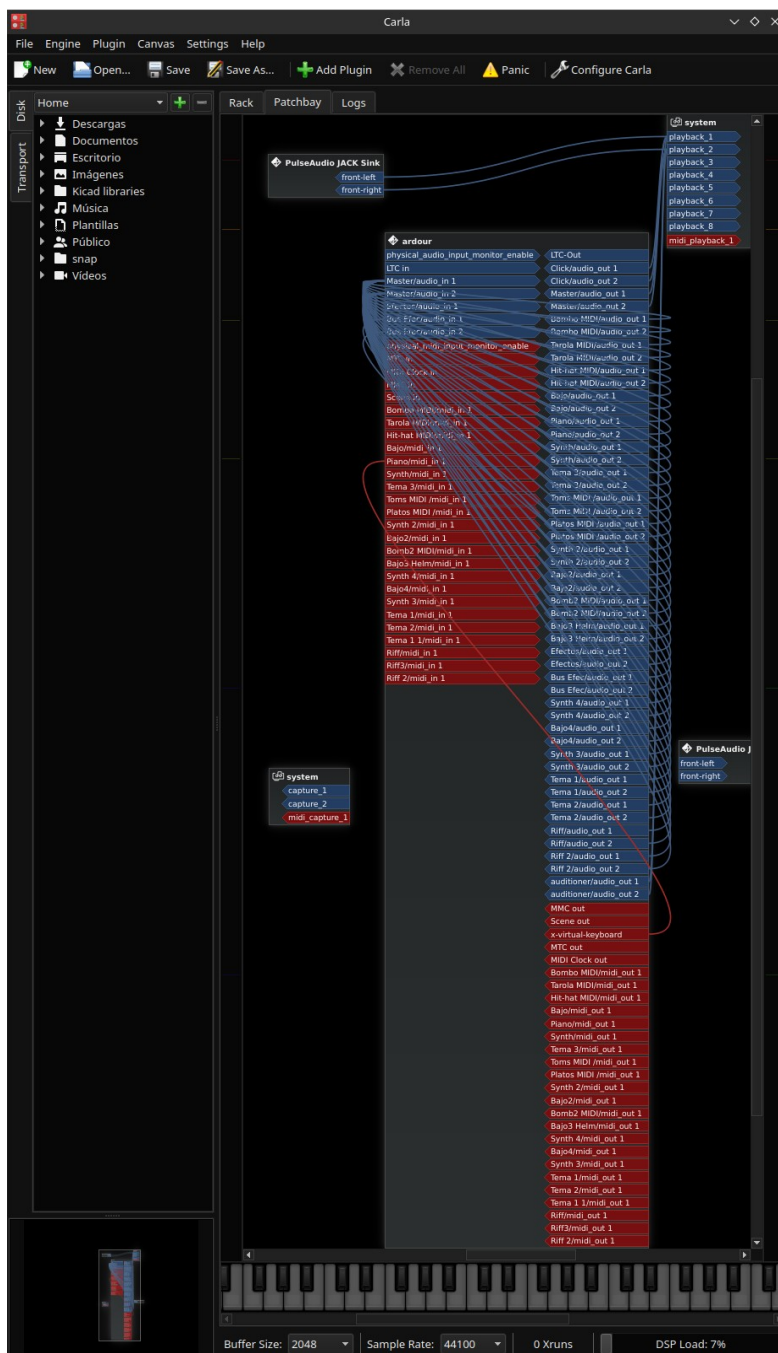
Pestaña de Rack en Carla.



En esta área se pueden cargar los plugins e instrumentos virtuales de manera externa al DAW. Se cargó un sintetizador, ecualizador y un *reverb* para ser manipulado por un controlador MIDI, la señal procesada mediante estos plugins puede ser ruteada a cualquier programa de audio que lo permita en la pestaña de *Patchbay*.

Figura 18

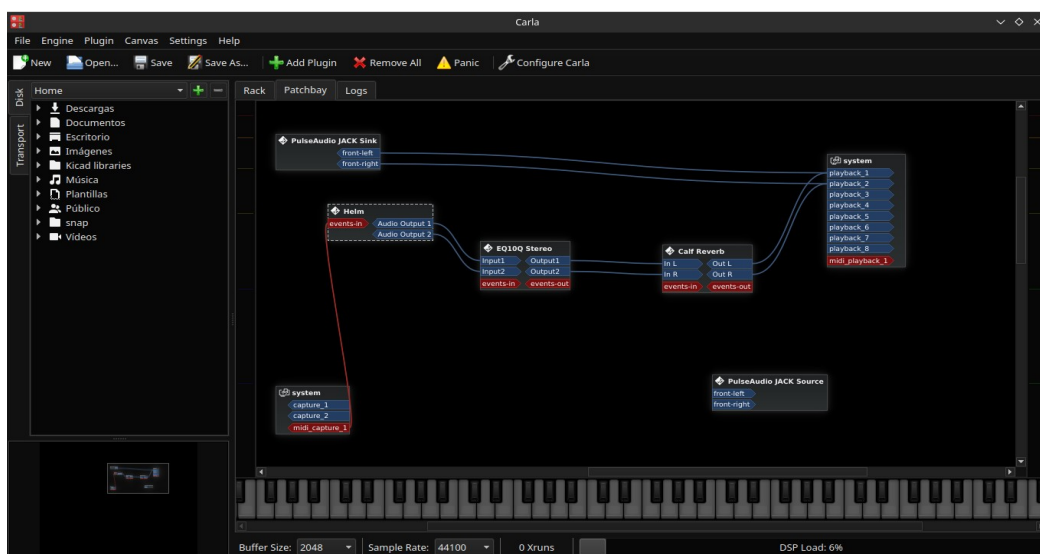
Presentación de la pestaña de Patchbay de Carla.



En la imagen anterior se pueden ver todas las conexión que hay en la sesión cargada del DAW. En esta área se pueden manipular todas las conexiones y afectan a los programas que estén haciendo uso de estas.

Figura 19

Patchbay de Carla.



En el *Patchbay* se gestionan todas las entradas, salidas de los plugins y programas que estén haciendo uso de Carla y QjackCtl, el color azul representa el flujo de la señal de audio que puede ser generada mediante un instrumento, audio pregrabado o instrumentos virtuales y en rojo las señal MIDI que generar un controlador o secuenciador MIDI. En Carla igual se pueden crear sesiones que permiten ahorrar tiempo a la hora de hacer conexiones y no tener que cargar todo desde un punto cero.

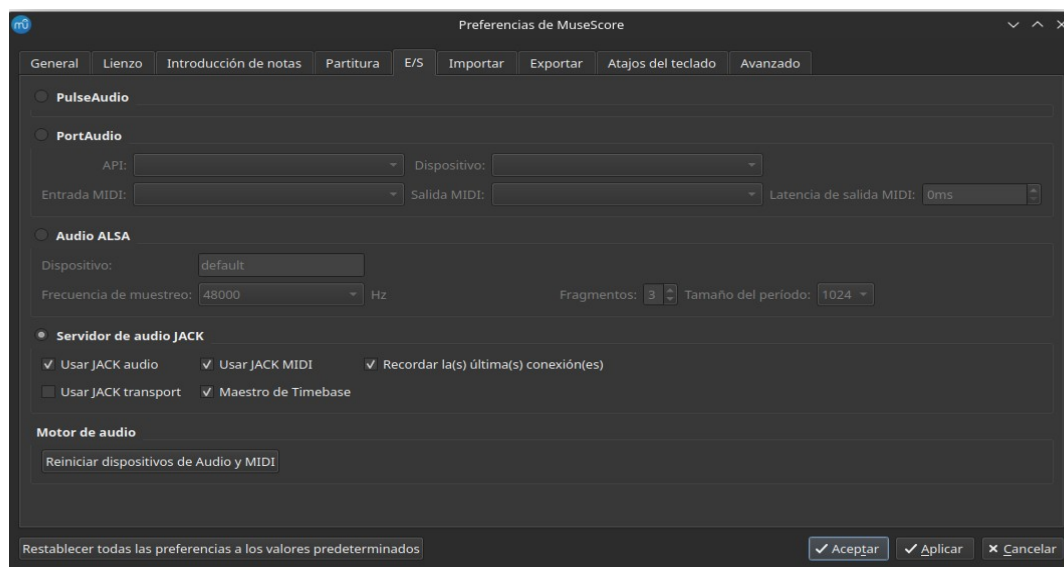
Musescore

Musescore es por defecto uno de los editores de partituras más usados en las distribuciones GNU/Linux y permite el uso del estándar MIDI, permitiendo gestionar las señales de audio y encadenarlo a otros plugins externos siempre y cuando el motor de E/S esté

configurado para funcionar junto a QjackCtl, esto permite tener una valoración más objetiva de la producción musical.

Figura 20

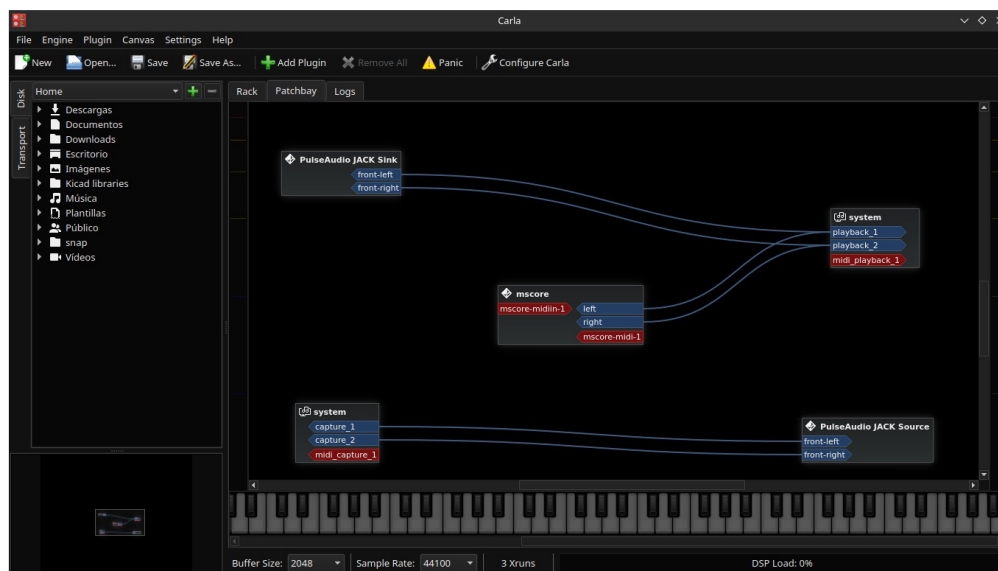
Parámetros de configuración MuseScore.



Una parte muy interesante de MuseScore es la posibilidad de poder gestionar el motor E/S para la distribución Ubuntu Studio que igual aplica para otras distribuciones GNU/Linux. Es importante tener una configuración correcta en QjackCtl para tener una estabilidad óptima en el equipo y evitar que el software se congele.

Figura 21

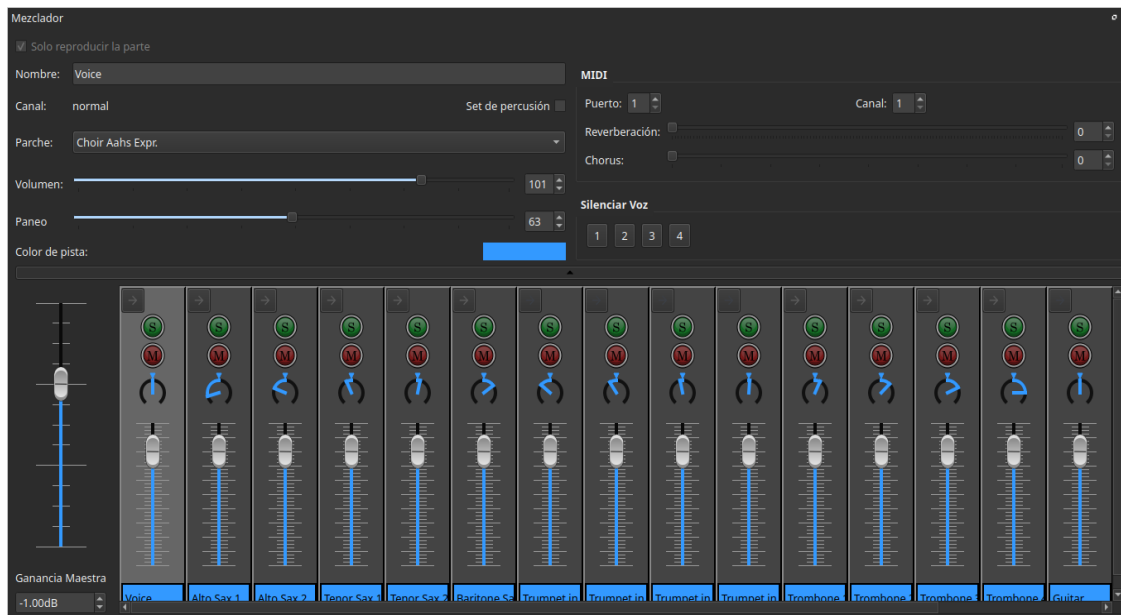
Muscore bajo el control de QjackCtl.



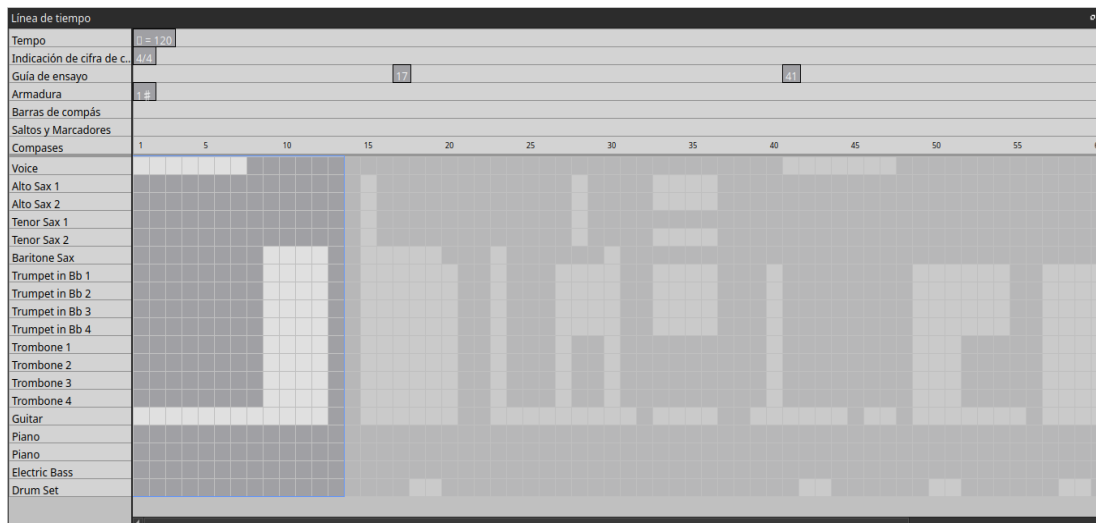
Cuando los software de audio son configurados de la forma correcta y trabajan en conjunto con QjackCtl y Carla, estos aparecen en el Patchbay de la siguiente forma, Muscore aparece en el centro de la imagen como un rectángulo que refleja sus entradas y salidas tanto de audio como MIDI . En la imagen las E/S se representan en colores azul para el audio y rojo para el MIDI.

Figura 22

Mezclador de MuseScore.



Una de las herramientas más importantes para un editor de partituras es la posibilidad de hacer las mezclas básicas del audio. MuseScore incorpora un mezclador de audio más que básico incorporando *faders* de ganancia, paneo, botones de solo y muteo por cada instrumento y permite manipular cada instrumento gracias al protocolo MIDI que permite cargar paquetes Soundfonts con extensión sf2 o sf3 y el formato SFZ, para dotar las composiciones de un realismo lo más cercano a un instrumento real, mejorando la referencia auditiva del trabajo. Todos los parámetros son manipulables permitiendo un trabajo más exacto y con mejor contenido.

Figura 23*Línea de tiempo.*

Aquí permite desplazarse más rápido por los compases que se representan en pequeños cuadros, de igual forma se pueden ver las diferentes indicaciones de la partitura que funcionan como puntos de desplazamiento. Estas herramientas se pueden desacoplar para poder ser enviadas a otra pantalla si la computadora está configurada para trabajar con escritorio extendido.

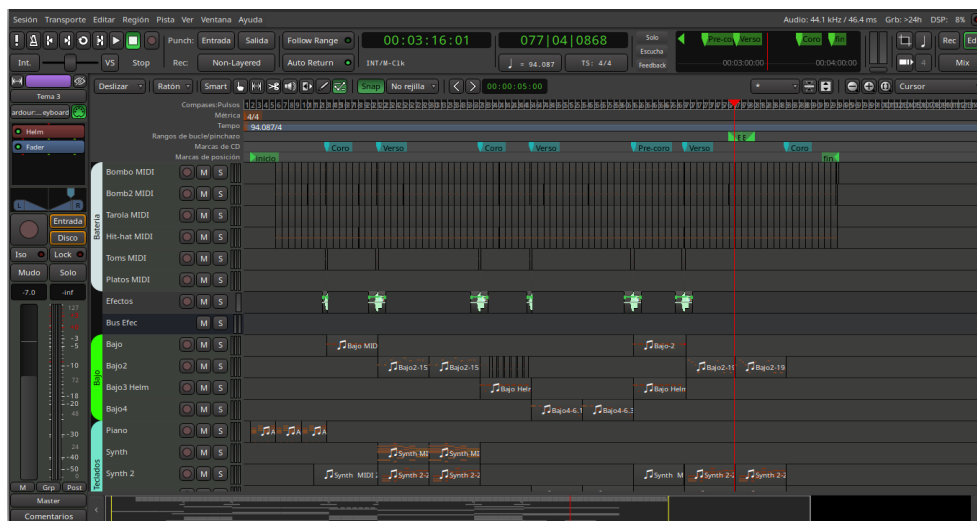
Ardour

Ardour es el *DAW* más avanzado disponible para las distribuciones GNU/Linux y que al combinarlos con herramientas como QjackCtl y Carla se logra un control óptimo de las E/S del software al trabajar. Por defecto soporta pistas de audio ilimitadas, ventanas específicas para rutear la señal sin salir del software, soporte para los múltiples formatos de audio.

En la figura 24 se muestra la primera área de trabajo de *Ardour* que permite manipular las pistas de audio y MIDI, esto no es muy diferente de otros software similares ya que las herramientas que incorporan son la base de la edición de audio.

Figura 24

Área de trabajo de ardour.



Otras de las áreas importantes se puede ver en la figura 25, la ventana del mezclador que nos permite ver la sesión como si de una mixer se tratara, en esta área podremos ajustar la ganancia de cada pista, ajustar el panning, hacer ruteos e insertar plugins para modificar el color de cada pista.

Figura 25

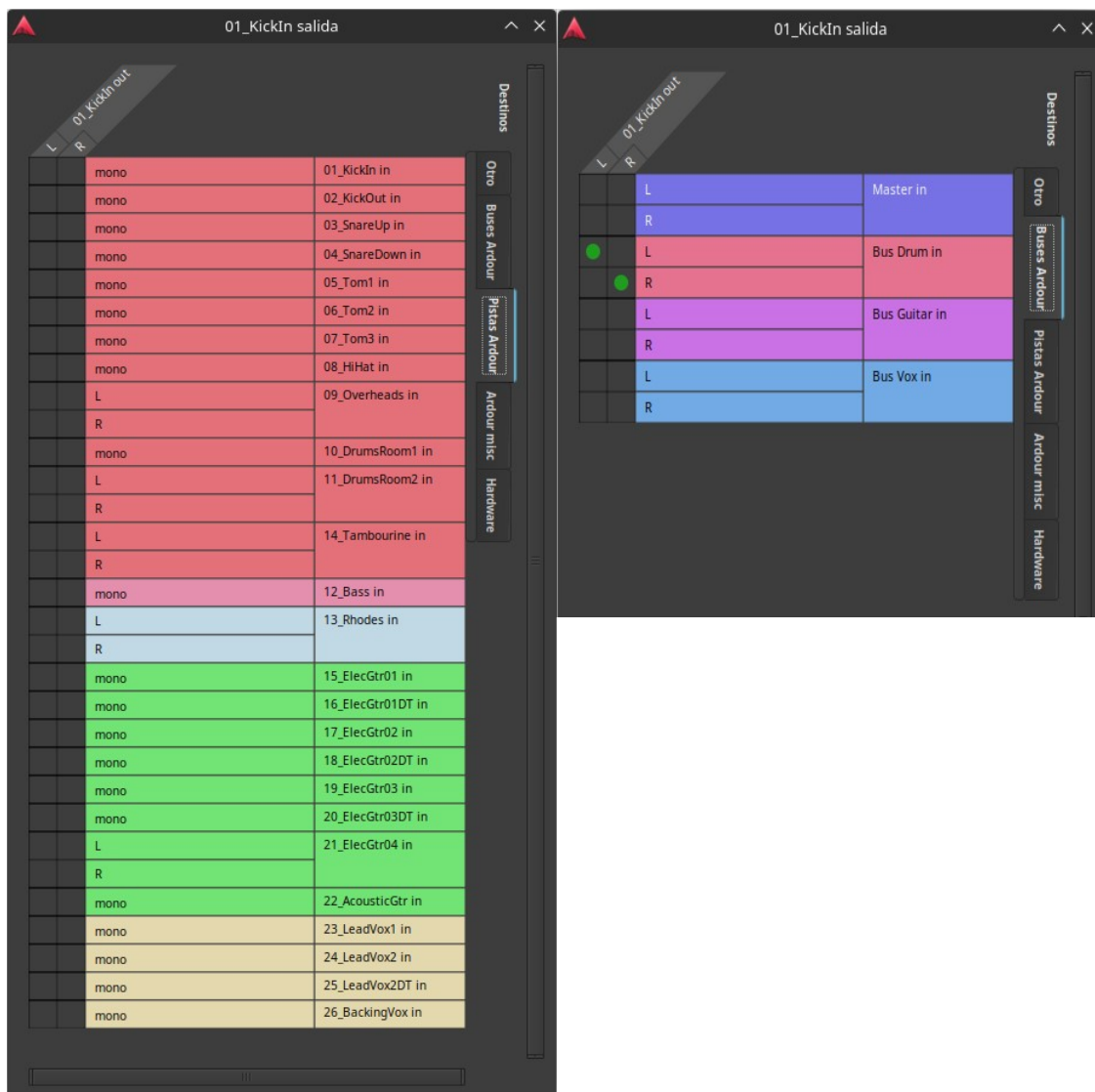
Área de Mezcla.



El área de ruteo es muy importante ya que es donde podremos redireccionar las señales a diferentes puntos de la sesión que estemos trabajando, a como se muestra en la figura 26.

Figura 26

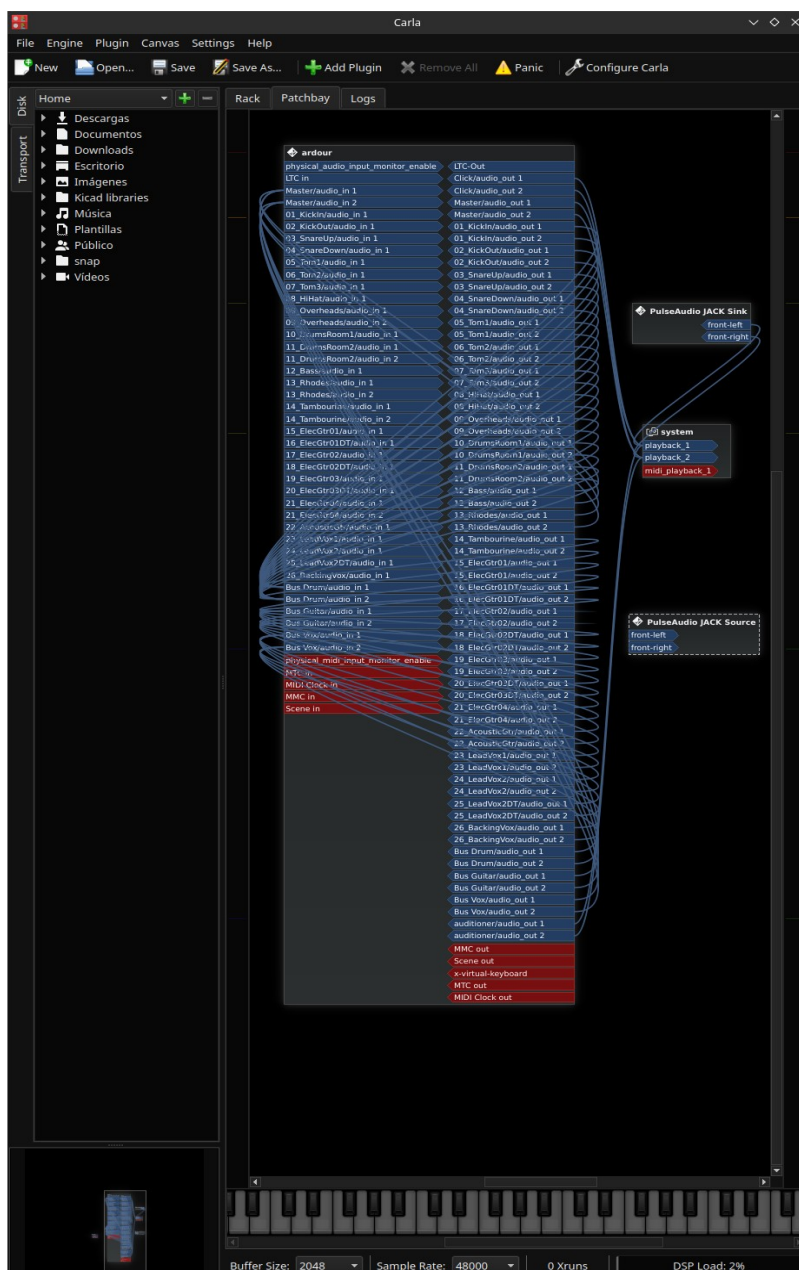
Ventana de ruteo para señales de audio y MIDI.



Todo lo que se ve en el área de ruteo del DAW también se puede ver en Carla en la pestaña de *Patchbay* pero de forma más visual como se puede ver en la figura 27 y con la posibilidad de desviar esas señales y pasarla por un procesador externos y retornarla al DAW.

Figura 27

Como se ven los ruteos del DAW en Carla.



Algo muy importante de Ubuntu Studio junto al DAW es la colección de plugins de diversos autores como: Alexander Ehlert, Andy Wingo, Ardour Community, Calf Studio Gear, Damien Zammit, David Robillard, Filipe Coelho, Invada, Linux Studio Plugins LSP, Mike Rawes, Pere Rafols Soler, Steve Harris y Tim Goetze, que vienen preinstalados y que se pueden ejecutar mediante Carla, Ardour y otros software que permitan ejecutar estos plugins. Existen plugins de marcas privadas como: Overtone DSP, u-he y Tonelib que a día de hoy son muy pocas, con poco soporte pero funcionales y que conforme el mercado crece por el uso de estos sistemas GNU/Linux las marcas de prestigio no dudaran en portear sus software a esta plataforma. En las figuras 28 y 29 se pueden ver la lista de algunos de los plugins preinstalados.

Figura 28

Lista de plugins.

Fav	Nombre	Tags	Creador
<input type="checkbox"/>	TAP AutoPanner	modulation	Tom Szilagyi
<input type="checkbox"/>	TAP Chorus/Flanger	modulation	Tom Szilagyi
<input type="checkbox"/>	TAP DeEsser	dyn:de-esser	Tom Szilagyi
<input type="checkbox"/>	TAP Dynamics (M)	dyn:compressor dyn:gate/expander...	Tom Szilagyi
<input type="checkbox"/>	TAP Dynamics (St)	dyn:compressor dyn:gate/expander...	Tom Szilagyi
<input type="checkbox"/>	TAP Equalizer	eq:parametric	Tom Szilagyi
<input type="checkbox"/>	TAP Equalizer/BW	eq:parametric	Tom Szilagyi
<input type="checkbox"/>	TAP Fractal Doubler	modulation pitch-shift	Tom Szilagyi
<input type="checkbox"/>	TAP Pitch Shifter	modulation pitch-shift	Tom Szilagyi
<input type="checkbox"/>	TAP Reflector	delay	Tom Szilagyi
<input type="checkbox"/>	TAP Reverberator	reverb	Tom Szilagyi
<input type="checkbox"/>	TAP Rotary Speaker	modulation	Tom Szilagyi
<input type="checkbox"/>	TAP Scaling Limiter	dyn:limiter	Tom Szilagyi
<input type="checkbox"/>	TAP Sigmoid Booster	distortion	Tom Szilagyi
<input type="checkbox"/>	TAP Stereo Echo	delay	Tom Szilagyi
<input type="checkbox"/>	TAP Tremolo	modulation	Tom Szilagyi
<input type="checkbox"/>	TAP TubeWarmth	distortion	Tom Szilagyi
<input type="checkbox"/>	Vocoder	distortion	Achim Settelmeier
<input type="checkbox"/>	Glame Bandpass Analog Filter	eq:filter	Alexander Ehlert
<input type="checkbox"/>	Glame Bandpass Filter	eq:filter	Alexander Ehlert
<input type="checkbox"/>	GLAME Butterworth Highpass	eq:filter	Alexander Ehlert
<input type="checkbox"/>	GLAME Butterworth Lowpass	eq:filter	Alexander Ehlert
<input type="checkbox"/>	Glame Butterworth X-over Filter	eq:filter utility	Alexander Ehlert
<input type="checkbox"/>	Glame Highpass Filter	eq:filter	Alexander Ehlert
<input type="checkbox"/>	Glame Lowpass Filter	eq:filter	Alexander Ehlert
<input type="checkbox"/>	Mag's Notch Filter	eq:filter	Alexander Ehlert
<input type="checkbox"/>	Allpass delay line, cubic spline interpolation	example	Andy Wingo
<input type="checkbox"/>	Allpass delay line, linear interpolation	example	Andy Wingo
<input type="checkbox"/>	Allpass delay line, noninterpolating	example	Andy Wingo
<input type="checkbox"/>	Comb delay line, cubic spline interpolation	example	Andy Wingo
<input type="checkbox"/>	Comb delay line, linear interpolation	example	Andy Wingo
<input type="checkbox"/>	Comb delay line, noninterpolating	example	Andy Wingo
<input type="checkbox"/>	Simple delay line, cubic spline interpolation	example	Andy Wingo
<input type="checkbox"/>	Simple delay line, linear interpolation	example	Andy Wingo

Figura 29

Lista de plugins (continuación).

Fav	Nombre	Tags	Creador
<input type="checkbox"/>	Calf Equalizer 8 Band	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Exciter	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Filter	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Filterclavier	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Flanger	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Gate	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Haas Stereo Enhancer	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Limiter	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Mono Compressor	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Mono Input	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Multi Chorus	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Multi Spread	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Multiband Compressor	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Multiband Enhancer	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Multiband Gate	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Multiband Limiter	plugin	Calf Studio Gear
<input type="checkbox"/>	Calf Phaser	plugin	Calf Studio Gear
<input type="checkbox"/>	LSP Gate LeftRight	plugin	LSP LV2
<input type="checkbox"/>	LSP Gate MidSide	plugin	LSP LV2
<input type="checkbox"/>	LSP Gate Mono	plugin	LSP LV2
<input type="checkbox"/>	LSP Gate Stereo	plugin	LSP LV2
<input checked="" type="checkbox"/>	LSP Graphic Equalizer x16 LeftRight	plugin	LSP LV2
<input type="checkbox"/>	LSP Graphic Equalizer x16 MidSide	plugin	LSP LV2
<input type="checkbox"/>	LSP Graphic Equalizer x16 Mono	plugin	LSP LV2
<input type="checkbox"/>	LSP Graphic Equalizer x16 Stereo	plugin	LSP LV2
<input type="checkbox"/>	LSP Graphic Equalizer x32 LeftRight	plugin	LSP LV2
<input type="checkbox"/>	LSP Graphic Equalizer x32 MidSide	plugin	LSP LV2
<input type="checkbox"/>	LSP Graphic Equalizer x32 Mono	plugin	LSP LV2
<input type="checkbox"/>	LSP Graphic Equalizer x32 Stereo	plugin	LSP LV2
<input type="checkbox"/>	LSP Impulse Responses Mono	plugin	LSP LV2
<input type="checkbox"/>	LSP Impulse Responses Stereo	plugin	LSP LV2
<input type="checkbox"/>	LSP Impulse Reverb Mono	plugin	LSP LV2
<input type="checkbox"/>	LSP Impulse Reverb Stereo	plugin	LSP LV2
<input type="checkbox"/>	LSP Latency Meter	plugin	LSP LV2

Las posibilidades de este DAW son ilimitadas, compitiendo con los más profesionales y permitiéndoles a usuarios que están empezando a conocer este conjunto de herramientas sin ninguna limitante hasta el punto de poder hacer trabajos profesionales, teniendo plugins de buena calidad que cubren todas las necesidades para este tipo de trabajos.

Sonic Visualiser

Este software permite visualizar y analizar pistas de audio para su estudio, teniendo como herramientas básicas la posibilidad de bajar o aumentar la velocidad sin alterar el tono, crear regiones, bucle, otras herramientas más avanzadas para laboratorios como visor de forma de onda, espectrograma básico, espectrograma de rango melódico, espectrograma de frecuencia pico, analizador de espectro audible y el soporte para plugins externos.

Figura 30

Presentación de Sonic Visualiser.

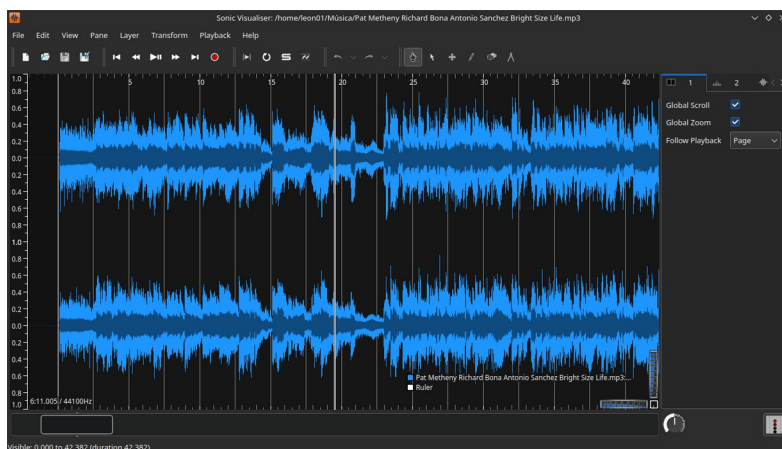


Figura 31

Analizador de espectro audible.

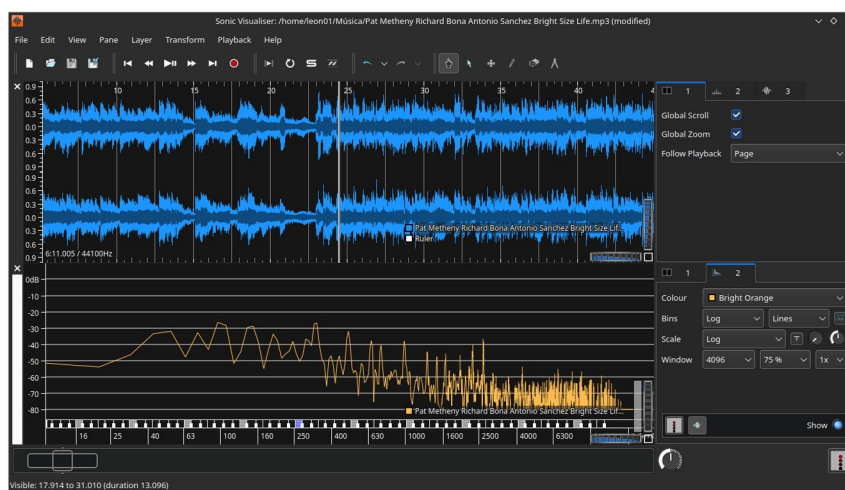
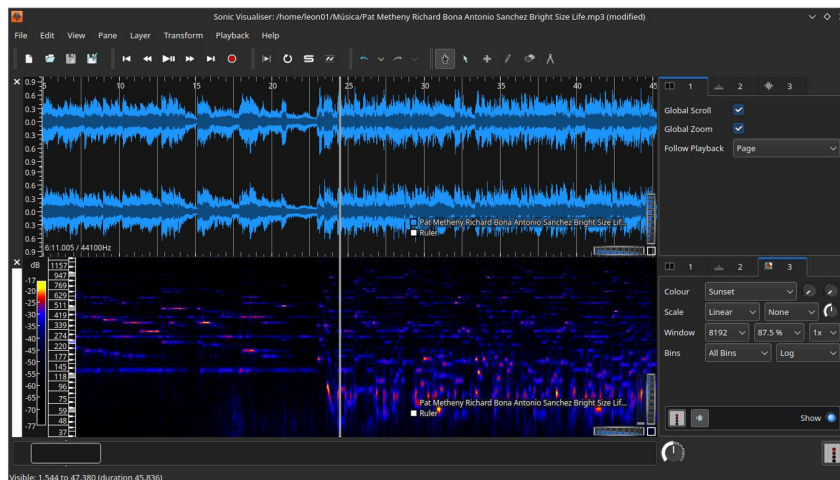


Figura 32

Espectrograma de rango melódico.



Este software le permite a los estudiantes e interesados usarlo como un reproductor para poder hacer transcripciones y análisis de pistas de audio de todo tipo.

Lo Importante del Software sin Limitaciones

En este capítulo se comprueba que para trabajar en producción musical y audio de forma profesional lo importante es tener un software sin limitaciones, que utilice bien los recursos del hardware que una computadora es capaz de brindar dependiendo sus capacidades y para los iniciados en estas materias, es esencial tener todas las herramientas disponibles para sus proyectos y Ubuntu Studio cumple con todas ellas siendo una alternativas eficaz libres y confiables para todos estos trabajos y con una comunidad que ha hecho que este sistema operativo pueda adaptar a las tecnologías de la industria musical, en la cual las instituciones, alumnos, personas y empresas pueden aportar para su mantenimiento con diversos tipos de ayuda.

Conclusiones

La evolución de la escritura musical a través de la historia es una muestra de su adaptabilidad hacia las nuevas tecnologías, pasando del manuscrito a la música impresa y llegar hasta nuestros días a la edición digital a través de las computadoras, teniendo en cuenta todas las alternativas que brinda el software libre para la creación, investigación y la producción musical.

Como experiencia durante todo el curso de la Licenciatura de Jazz y Música Popular, pude utilizar y poner a prueba el software privado y software libre; por lo anterior estoy convencido de que esta última tecnología rivaliza y supera el software privado, debido a sus costos, fácil accesibilidad, como una alternativa más y por el concepto de libertad para que el usuario final pueda mejorar, compartir y conocer las nuevas tecnologías sin las restricciones habituales, donde los usuarios que dan soporte al software libre son de todo el mundo y han sabido organizarse intelectual y técnicamente para la elaboración de un proyecto común en beneficio del conocimiento.

A partir de la investigación realizada en el presente trabajo, se exponen los riesgos legales, daños a los equipos y robo de información personal que el artista corre al usar la piratería o el robo de la propiedad intelectual de algún software en específico para la creación, gestión y publicación de obras artísticas. Las alternativas de software libre nos exenta de tales riesgo, ya que brindan herramientas que pueden ser similares y, en ocasiones, incluso más capaces que sus contrapartes de software propietario, permitiendo también que los equipos duren más al tener un sistema que gestiona el rendimiento de manera óptima y sin descuidar las partes legales e intelectuales de dicha tecnología.

A partir de la presente investigación se puede ampliar su contenido para conocer el alcance, referencia, posibilidades de esta tecnología y sus futuras aplicaciones como alternativa en otras especialidades artísticas y científicas.

Referencias

- Avid (2018). *Acuerdo de licencia para el software de avid*. Pag. 219. <https://cdn-www.avid.com/-/media/avid/files/legal/01553009800-avid-eula-march-2018.pdf>
- Ableton. (s. f.). Try Ableton Live for free. *Ableton*. <https://www.ableton.com/en/trial/>
- ALSA project (Advanced Linux Sound Architecture). (s.f.). Advanced Linux Sound Architecture (ALSA) project homepage. https://www.alsa-project.org/main/index.php?title=Main_Page&oldid=7378
- Alcántara J. (s. f.). Licencia restrictiva. *Versvs*. <https://www.versvs.net/pedia/licencia-restrictiva/>
- Alonso R. (2020, 23 de Mayo). Comparativa Thunderbolt, ¿han merecido la pena sus mejoras?. *Hardzone*. <https://hardzone.es/reportajes/comparativas/thunderbolt-1-2-3-4-comparativa-versiones/>
- Andersen A. (2015, 16 de diciembre). The rise of reaper. *A sound effect*. <https://www.asoundeffect.com/the-rise-of-reaper/>
- Apple Inc. (2022). Vista previa de Mac App Store. *macOS Monterey*. <https://apps.apple.com/mx/app/mac-os-monterey/id1576738294>
- Avid Knowledge Base, (2021, 16 de Diciembre). Avid Knowledge Base. *Sibelius Operating System Compatibility Chart*. https://avid.secure.force.com/pkb/articles/en_US/Compatibility/Sibelius-Operating-System-Compatibility-Chart

Arobas Music. (s. f.). *Guitar Pro 7.6. Guitar Pro*. <https://www.guitar-pro.com/>

Ardour (s. f.). Who Did This?. *Ardour*. <https://ardour.org/credits.html>

Audacity. (s. f.). *Credits*. <https://www.audacityteam.org/about/credits/>

Avid Technology, Inc. (s. f.). <https://www.avid.com/es/pro-tools>

Blánquez J. (2016, 23 de Septiembre). Don Buchla, el otro padre del sintetizador. *El mundo*.

<https://www.elmundo.es/cultura/2016/09/23/57e4eb61ca4741c21a8b458f.html>

Barahona. J. G. (2012. 23 de Mayo). *Licencias permisivas en Software libre*. EOI.

<https://www.eoi.es/wiki/index.php?>

[title=Licencias_permisivas_en_Software_libre&oldid=8467](https://www.eoi.es/wiki/index.php?title=Licencias_permisivas_en_Software_libre&oldid=8467)

Brocca J. C. y Casamiquela R. (2005). Las licencias de software desde la perspectiva del usuario final. *Revista Pilquen*, Sección Ciencias Sociales. No. 7.

<http://www.redalyc.org/articulo.oa?id=347532051012>

Canonical. (s.f.). The story of Ubuntu. *Canonical*. <https://ubuntu.com/about>

Cespedes I. (2014, 23 de Marzo). Plugins para GNU/Linux: entendiendo sus diferencias.

Hispasonic. <https://www.hispasonic.com/reportajes/plugins-para-gnu-linux-entendiendo-diferencias/39124>

Universidad de Chile. (2018, 20 de Agosto). *Richard Stallman en Beauchef: "Hemos construido este muro de libertad en la informática"*. Universidad de Chile.

<https://ingenieria.uchile.cl/noticias/146197/richard-stallman-realiza-charla-en-beauchef>

Cockos Incorporated. (s. f.). <https://cockos.com/>

Cruz, J. (2019, 21 de Enero). Consecuencias del uso ilegal de software. *IdcOnline*.

<https://idconline.mx/corporativo/2019/01/21/consecuencias-del-uso-ilegal-de-software>

Salces, D. (2021, 25 de Marzo). *Mac OS X cumple 20 años: damos un repaso a su historia*.

Muy computer. <https://www.muycomputer.com/2021/03/25/mac-os-x-cumple-20-anos/>

Davis P. (2019, Diciembre). *Ardour: 20th birthday*. Ardour. [https://discourse.ardour.org/t/ardour-](https://discourse.ardour.org/t/ardour-20th-birthday/102333)

[20th-birthday/102333](https://discourse.ardour.org/t/ardour-20th-birthday/102333)

Download. (s. f.). Download Audacity. *Audacity*. <https://www.audacityteam.org/download/>

Dominguez C. y Rigueiro J. (2015). Carta de Guido dirigida al monje Miguel acerca de un canto

desconocido. (1a). <https://www.aacademica.org/jorge.rigueiro.garcia/10.pdf>

Dssi. (s. f.). *What is DSSI?*. Dssi. <http://dssi.sourceforge.net/>

Futuremusic. (2014, 07 de Mayo). *Cubase, la historia del gran DAW de Steinberg*. *Futuremusic*.

<http://www.futuremusic-es.com/steinberg-cumple-30-anos-breve-historia-de-cubase/>

Fernández Y. (2020, 17 de Junio). FireWire: qué es, qué tipos hay y cuáles son sus

características. *Xataka*. [https://www.xataka.com/basics/firewire-que-que-tipos-hay-](https://www.xataka.com/basics/firewire-que-que-tipos-hay-cuales-sus-caracteristicas)

[cuales-sus-caracteristicas](https://www.xataka.com/basics/firewire-que-que-tipos-hay-cuales-sus-caracteristicas)

Fernández Y. (2021, 21 de Abril). Tipos de USB: estándares, conectores y características de

cada uno. *Xataka*. [https://www.xataka.com/basics/tipos-usb-estandares-conectores-](https://www.xataka.com/basics/tipos-usb-estandares-conectores-caracteristicas-cada-uno)

[caracteristicas-cada-uno](https://www.xataka.com/basics/tipos-usb-estandares-conectores-caracteristicas-cada-uno)

Free Software Foundation. (2022, 04 de Febrero). *¿Qué es GNU?*. Free Software Foundation.
<https://www.gnu.org/home.es.html>

Free Software Foundation. (s. f.). *Categorías de software libre y software que no es libre*. Free Software Foundation. <https://www.gnu.org/philosophy/categories.es.html>

Free Software Foundation. (1989. 1 de Febrero). *GNU General Public License, version 1*. Free Software Foundation. <https://www.gnu.org/licenses/old-licenses/gpl-1.0.html>

Free Software Foundation. (1991. 2 de Junio). *GNU Library General Public License, version 2.0*. Free Software Foundation. <https://www.gnu.org/licenses/old-licenses/lgpl-2.0.html>

Free Software Foundation. (1999. Febrero). *GNU Lesser General Public License, version 2.1*. Free Software Foundation. <https://www.gnu.org/licenses/old-licenses/lgpl-2.1.html>

Free Software Foundation. (2007. 29 de Junio). *GNU Lesser General Public License*. Free Software Foundation. <https://www.gnu.org/licenses/lgpl-3.0.html>

Free Software Foundation. (2007. 3 de Junio). *GNU General public license*. Free Software Foundation. <https://www.gnu.org/licenses/gpl-3.0.html>

Future Music. (2011, 30 de mayo). *A brief history of Pro Tools*. Musicradar.
<https://www.musicradar.com/tuition/tech/a-brief-history-of-pro-tools-452963>

Future Music. (2011, 13 de enero). *A brief history of Ableton Live*. Musicradar.
<https://www.musicradar.com/tuition/tech/a-brief-history-of-ableton-live-357837>

Gauzak. (2017, 25 de Abril). *Ikutaro Takehashi (1930-2017)*. Gauzak.

<https://www.gauzak.com/ikutaro-takehashi/>

Ardour. (s. f.). *Get Ardour*. Ardour. [https://community.ardour.org/download?](https://community.ardour.org/download?platform=linux&architecture=x86_64&type=compiled)

[platform=linux&architecture=x86_64&type=compiled](https://community.ardour.org/download?platform=linux&architecture=x86_64&type=compiled)

Gemtracks. (2021, 31 de diciembre). *Pro Tools Complete History + Versions (2022 UPDATED)*.

Gemtracks. <https://www.gemtracks.com/guides/view.php?title=pro-tools-history&id=1594>

González G. (s. f.). *Open source: copyleft vs licencias permisivas*. Thinkbig.

<https://blogthinkbig.com/licencias-open-source>

González J. M. (2021). *Apuntes para una historia de la música*. (1ra ed).

<http://hdl.handle.net/10201/102321>

Gutiérrez, C. (2013, 13 de Septiembre). *Cracks y los riesgos para la seguridad*.

Welivesecurity. <https://www.welivesecurity.com/la-es/2013/09/13/cracks-riesgos-seguridad/>

Haff G. (2019, 26 de Abril). *The mysterious history of the MIT License*. Opensource.

<https://opensource.com/article/19/4/history-mit-license>

Ibarra. B. (2016, Mayo). *Software Libre y Software Propietario. Software libre o software propietario para una efectiva transferencia de tecnología en Ecuador*. [Tesis, Universidad Internacional del Ecuador].

<https://repositorio.uide.edu.ec/handle/37000/1425>

Image Line. (2022, 11 de Marzo). *history*. Image Line. <https://www.image-line.com/history/>

Image Line. (s. f.). *Image line software end user license agreement*. Image Line.

<https://www.image-line.com/image-line-software-eula/>

Interpro. (2018, 30 de noviembre). *¿Cuáles son las diferencias entre licencia perpetua y suscripción?*. Interpro. <https://www.interpro.ec/faq/diferencias-licencia-perpetua-suscripcion/>

Internet Archive Wayback Machine. (1998, 03 de Diciembre). *Welcome to the official site of Guitar Pro for Windows 95*. Internet Archive Wayback Machine.

<https://web.archive.org/web/19981203060107/https://www.guitar-pro.com/>

Johnson T. (2009, 09 de Septiembre). *One Snapshot from Tom's Finale History Slideshow. the finale blog*. MakeMusic. <https://www.finalemusic.com/blog/one-snapshot-from-toms-finale-history-slideshow/>

Ladsp. (s. f.). *Linux Audio Developer's Simple Plugin API (LADSPA)*. Ladsp.

<https://www.ladspa.org/>

Logicbus (2019, 10 de julio). *Traslado de datos de automatización a través de interfaz USB*.

Logicbus. <https://www.logicbus.com.mx/blog/usb/>

López, P. (15 de Diciembre, 2019). *PulseAudio desmenuzamos el servidor de sonido de GNU/Linux*. Liberaturadio. <https://liberaturadio.org/desmenuzamos-el-control-de-volumen-de-pulseaudio/>

Luchetta E. (2020, 23 de Octubre). *La impresión de la música: historia y curiosidades de las partituras*. Pixartprinting. <https://www.pixartprinting.es/blog/impresion-musica-partituras/>

License. (2021, 16 de Noviembre). *Audacity License*. Audacity.

<https://manual.audacityteam.org/man/license.html>

Lilypond. (s. f.). *Downloads for LilyPond*. Lilypond. <http://lilypond.org/download.es.html>

Lilypond. (2002, julio). *GNU LilyPond*. Lilypond.

<http://lilypond.org/doc/v1.6/Documentation/user/out-www/lilypond/Preface.html#Preface>

The Linux Foundation. (s. f.). *The Linux Kernel Archives*. The Linux Foundation.

<https://www.kernel.org/>

MakeMusic. (s. f.). *Compre Software y Accesorios de MakeMusic*. Makemusic.

<https://store.makemusic.com/Store/default.aspx?tab=notation>

Martínez J. (2014, 05 de Febrero). *Polyphone, las soundfont más vivas que nunca*. *Informática musical*. Hispasonic. <https://www.hispasonic.com/reviews/polyphone-soundfont-mas-vivas-nunca/38973>

Marco M. (2019, 22 de Noviembre). *Suscripción vs. Un licenciamiento perpetuo, qué conviene más*. Cisco Systems. <https://news-blogs.cisco.com/americas/es/2019/11/22/suscripcion-vs-un-licenciamiento-perpetuo-que-conviene-mas/>

Maymó P. (2013). *El ideario de lo sacro en Gregorio Magno (590-604). De los santos en la diplomacia pontificia*. [Tesis Doctoral, Univeritat de Barcelona].

https://diposit.ub.edu/dspace/bitstream/2445/46343/1/PMC_TESIS.pdf

Microsoft. (s. f.). *Facts About Microsoft*. Microsoft. <https://news.microsoft.com/facts-about-microsoft/>

Moreno R. (2013, 25 de Mayo). *Historia del MIDI*. Diffusion Magazine.

<https://www.diffusionmagazine.com/index.php/biblioteca/categorias/historia/335-historia-del-midi>

Moreno V. (2014, 20 de Septiembre). *Biografía de Linus Torvalds*. Busca biografías.

<https://www.buscabiografias.com/biografia/verDetalle/8294/Linus%20Torvalds>

Música barroca, (s. f.) *Hisour Arte Cultura Historia*. Hisour. <https://www.hisour.com/es/baroque-music-30751/>

Music Printing History. (s. f.). *Finale (1988)*. Music Printing History.

<https://musicprintinghistory.org/finale/>

MuseScore. (2022). <https://musescore.org/es>

Microsoft Prensa. (2015, 6 de Abril). *Momentos destacados en la historia de Microsoft*.

Microsoft. <https://news.microsoft.com/es-es/2015/04/06/historia-microsoft-40-aniversario/>

O'Connor M. (2020, 21 de Julio). *Formatos de plugins: diferencias entre VST, VST3, AU, AAX, RTAS, TDM*. Electronic drumadvisor.

<https://www.electronicdrumadvisor.com/es/formatos-de-plugins-diferencias-entre-vst-vst3-au-aax-rtas-tdm/>

Oberheim. (s. f.). *Tom Oberheim*. Oberheim. <https://www.oberheim.com/about>

Polyphone. (2020). *Open source will win. Licenses*. Polyphone. <https://www.polyphone-soundfonts.com/license>

PreSonus. (s. f.). *PreSonus History*. PreSonus. <https://legacy.presonus.com/about/history>

PromocionMusical. (s. f.) *Historia de la Partitura*. Promociónmusical.

https://promocionmusical.es/historia-partitura#La_Partitura_y_La_Imprenta

Ranchal. J. (2014, 24 de Noviembre). *Microsoft confirma nuevo kernel 10 en Windows 10*. Muy computer. <https://www.muycomputer.com/2014/11/24/kernel-10-en-windows-10/>

Reaper. (2022, 5 de Marzo). *Download and Evaluate Reaper for Free*. Reaper.

<https://www.reaper.fm/download.php>

Red Hat. (2018, 19 de Marzo). *El Concepto de Linux*. Red Hat.

<https://www.redhat.com/es/topics/linux>

Rivera. A. S. (2020, 12 de Marzo). *Características de música en el Barroco*. Unprofesor.

<https://www.unprofesor.com/musica/caracteristicas-de-musica-en-el-barroco-4037.html>

Riquelme. R. (2019, 03 de Febrero). *Creative Commons tiene un nuevo capítulo en México*. El economista. <https://www.eleconomista.com.mx/tecnologia/Creative-Commons-tiene-un-nuevo-capitulo-en-Mexico-20190203-0002.html>

Roca. J. (2022, 28 de Julio). *USB, VGA, S/PDIF... Aprende a identificar todos los puertos de tu PC*. Hardzone. <https://hardzone.es/tutoriales/componentes/puertos-entrada-salida/>

Ruiz. I. (2006). *Iniciación a la historia de la música*.

<https://aulademusicamartinsarmiento.files.wordpress.com/2016/10/iniciaci3b3n-a-la-historia-de-la-mc3basica.pdf>

Sequential LLC. (2023). *About Dave Smith*. Sequential. <https://www.sequential.com/about-dave-smith/>

Sibelius Software, (s. f.). *Internet Archive Wayback Machine*,
<https://web.archive.org/web/20070527111154/http://www.sibelius.com/documents/index.html>

L & D, (2015. 14 de Mayo). *Software con dominio publico, Software semi-libre, Freeware y Shareware*. Wordpress.
<https://licencyandaplications.wordpress.com/2015/05/14/software-con-dominio-publico-software-semi-libre-freeware-y-shareware/>

Sourceforge. (2010, 21 de junio). *WYSIWYG music app makes a score*. Sourceforge.
<https://sourceforge.net/blog/wysiwyg-music-app-makes-a-score/>

Sfz format. (s. f.). *Welcome to SFZFormat.com!*. SFZ Format. <https://sfzformat.com/>

Stephen. J. (2021, Abril). *Sistema operativo Linux*. TechTarget.
<https://www.computerweekly.com/es/definicion/Sistema-operativo-Linux>

STmidiMusic. (s. f.). *Encore. GVOX-Encore*. <http://www.stmidimusic.com/html/encore.html>

Studio One. (s. f.). *Requisitos del sistema del ordenador*. Pre Sonus.
<https://www.presonus.com/es/studio-one-tech-specs.html>

Teresa. (s. f.). *La increíble historia de Apple*. A su servicio.net. <http://www.asuservicio.net/p/la-increible-historia-de-apple/>

The Bass Valley. (2021, 27 de Abril). *Breve Historia de los Sintetizadores*. The Bass Valley.

<https://thebassvalley.com/breve-historia-de-los-sintetizadores/>

The MIDI Association (s. f.). *MIDI History:Chapter 6-MIDI Is Born 1980-1983*. The MIDI

Association. <https://www.midi.org/midi-articles/midi-history-chapter-6-midi-begins-1981-1983>

Ubuntu Studio. (s.f.). *About*. Ubuntu Studio. <https://ubuntustudio.org/about/>

Velasco R. (2017, 02 de Octubre). *Apple apuesta por el OpenSource y publica el código del kernel de iOS y macOS en GitHub*. Redeszone.

<https://www.redeszone.net/2017/10/02/apple-opensource-kernel-ios-macos/>

Velduque. M. J. (2011). *El origen de la imprenta: la xilografía. La imprenta de Gutenberg*.

<file:///home/pc00/Descargas/Dialnet-ElOrigenDeLaImprentaLaXilografia-5169198.pdf>

Werner M. (1998). *El Arte de imprimir: de los tipos móviles al microprocesador*.

unesdoc.unesco.org/in/rest/annotationSVC/DownloadWatermarkedAttachment/attach_import_c64fa61e-c0cc-48f4-8cb0-98f1375d2b20?_=079609spao.pdf&to=36&from=1

Yoho S. (2015, 12 de junio). *Steven M. Alper and Finale History*. MakeMusic.

<https://www.finalemusic.com/blog/steven-m-alper-and-finale-history/>