

Componentes	Descripción				
Nombre del curso	Programación C++ de Audio Plugins				
Course Name	--				
Código					
Carácter	Electivo ambas especialidades				
Número de créditos SCT	6 créditos SCT (4,5 horas semanales - 81 horas semestrales)				
		Hora de cátedra demostrativa presencial y directa con profesor	Horas de trabajo en taller y/o laboratorio con profesor (individual y/o grupal)	Horas de trabajo con ayudante (taller, laboratorio o clases de ejercicios)	Horas de trabajo autónomo del estudiante (individual y/o grupal)
	Semanal	1,5	1,5		1,5
	Semestral	27,0	27,0		27,0
Línea de Formación	Ambas menciones				
Nivel	8vo Semestre, 4to Año				
Requisitos deseables	Álgebra Lineal, Cálculo Diferencial y Programación.				
Propósito formativo	<p>El estudiante comprenderá la teoría y los algoritmos computacionales para el diseño y compilación de plugins de audio para producción musical y audiovisual, usando el lenguaje de programación C++ y el framework JUCE.</p> <p>El curso tendrá un enfoque teórico-práctico que tiene como propósito desarrollar aplicaciones computacionales en C++ y JUCE que permitan procesar y controlar gráficamente (GUI) señales de audio digital en tiempo real.</p>				
Competencias específicas a las que contribuye el curso	<p><i>Competencia 1.3: Crear e intervenir los elementos constitutivos de la abstracción sonora que forman parte de una expresión artística a través de los procesos de codificación, generación, transmisión y recepción de la energía sonora de manera intencionada y reflexiva.</i></p> <p><i>Sub-Competencia 1.3.1: Procesando digitalmente la señal sonora a fin de preservarla y/o transformarla de manera intencionada.</i></p> <p><i>Competencia 1.4: Diseñar e implementar de forma unificada y coherente sistemas electroacústicos tanto en base a hardware preexistente y herramientas de desarrollo de software para aplicaciones profesionales y/o investigación.</i></p> <p><i>Sub-Competencia 1.4.1: Implementando y planificando sistemas de audio en base a hardware y software, para</i></p>				

	<p><i>aplicaciones profesionales y/o de investigación.</i></p> <p><i>Competencia 3.4: Analizar auditivamente el sonido, desde la perspectiva técnico-científica y desde su construcción artística</i></p> <p><i>Sub-Competencia 3.4.1: Reconociendo diversos fenómenos sonoros desde el punto de vista científico, artístico y tecnológico.</i></p> <p><i>Sub-Competencia 3.4.2: Evaluando fenómenos sonoros, emitiendo juicios de valor que le permitan tomar las decisiones pertinentes.</i></p> <p><i>Competencia 4.2: Proponer, administrar, operar y los recursos tecnológicos y materiales, que permitan que los proyectos asociados al área del sonido se concreten adecuadamente.</i></p> <p><i>Sub-Competencia 4.2.1: Evaluando e interpretando los requerimientos estéticos y técnicos para proponer soluciones pertinentes.</i></p> <p><i>Sub-Competencia 4.2.2: Desarrollando habilidades operativas que le permitan un manejo de las herramientas tecnológicas, de manera eficiente, ordenada y sistematizada.</i></p> <p><i>Competencia 4.4: Reflexionar y proponer una sonoridad coherente con la intención del proyecto y/o obra artística involucrada, mediante el uso creativo y funcional de herramientas tecnológicas</i></p> <p><i>Sub-Competencia 4.4.3: Escogiendo las herramientas más adecuados asociados a la intencionalidad del proyecto y/o obra artística, a partir de los recursos existentes.</i></p>
Resultados de aprendizaje	<p>Al final de esta actividad curricular el estudiante será capaz de:</p> <ul style="list-style-type: none"> ● Desarrollar una comprensión global de los fundamentos de programación C++ y Juce. ● Reconocer las estructuras de sistemas discretos esenciales para distintos procesos de audio digital. ● Implementar en C++ algoritmos de procesamiento y análisis de audio digital en tiempo real. ● Diseñar y enlazar una interfaz gráfica (GUI) para el control en tiempo real del plugin de audio..
Saberes /	<p>Unidad I: Fundamento de lenguaje C y C++ en ambiente Windows</p> <ul style="list-style-type: none"> ● Tipos de datos ● Control de flujo

<p>Contenidos</p>	<ul style="list-style-type: none"> ● Punteros ● Funciones y Clases ● Herencia y mutabilidad ● Instrucciones al compilador <p>Unidad II: Framework JUCE en ambiente Windows</p> <ul style="list-style-type: none"> ● Características y soporte multiplataforma (OSX y Windows) ● Librerías de Juce ● Creación de proyectos y compilación ● Interfaz gráfica de usuario <p>Unidad III: Diseño de Plugins de Audio Teoría y desarrollo de los siguientes plugins esenciales:</p> <ul style="list-style-type: none"> ● Ganancia ● Retardo Unitario ● Filtros ● Procesamiento de Dinámica: Compresión
<p>Metodologías</p>	<p>Clases demostrativas, con desarrollo de problemas y demostraciones de casos. Actividades de prácticas guiadas para los estudiantes mediante el uso de herramientas computacionales.</p>
<p>Evaluación</p>	<p>La evaluación consistirá en:</p> <p>2 Trabajos de programación individuales (25% c/u) 2 Quizzes Individuales (10% c/u) 1 Proyecto Final Individual (30%)</p>
<p>Requisitos de aprobación</p>	<p>Para aprobar el curso el estudiante debe tener una nota final superior o igual a cuatro punto cero. De acuerdo a la fórmula: Nota Final = Nota de Presentación *60% + Nota Examen * 40%</p> <p>Condiciones específicas, indicadas en Reglamento de Facultad.</p>
<p>Palabras clave</p>	<p>Lenguaje c, Lenguaje cpp, Audio Effects, DAFX, Filters, FFT, Fourier.</p>
<p>Bibliografía</p>	<ul style="list-style-type: none"> ● Designing Audio Effect Plugins in C++. Authored by Will C. Pirkle. Ed. Focal Press. 2019.

	<ul style="list-style-type: none"> ● Audio Effects: Theory, Implementation, and Application. Authored by Joshua D. Reiss & Andrew McPherson. CRC Press. 2015. ● Physical Audio Signal Processing: for Virtual Musical Instruments and Digital Audio Effects. Authored by Julius Smith. W3K Publishing (December 21, 2010). Book online: https://ccrma.stanford.edu/~jos/pasp/ (link visto en Julio 2019). ● Sophocles J. Orfanidis, Introduction to Signal Processing, 2010, Disponible en http://www.ece.rutgers.edu/~orfanidi/intro2sp/ (link visto en Julio 2019).
Recursos Complementarios	<ul style="list-style-type: none"> ● Matlab (Octave) Software para simulaciones. ● Steven W. Smith, The Scientist and Engineer's Guide to Digital Signal Processing, California Technical Pub; 1st edition, 1997. Disponible en http://www.dspguide.com (link visto en Julio 2019). ● Visual Studio 17, Microsoft Inc. ● JUCE 7, https://juce.com/

Autores del programa: Prof. Víctor M. Espinoza (Ph.D.) y Prof. Ricardo Saldías (M.Sc.)