



DIRECCIÓN DE FORMACIÓN HUMANISTA Y CRISTIANA

Programa de formación para la diversidad

Programa de Curso

1. Nombre del curso		Pensamiento computacional y programación para el buen vivir					
2. Código		DDO1118					
3. Año /Semestre		4° a 8° semestre - Todas las carreras					
4. Créditos SCT-Chile		3 SCT PMA 1-1-3					
Nº Horas PMA Total	84	Horas Presenciales	17	Horas Mixtas	17	Horas Autónomas	50
5. Tipo de Curso		<input type="checkbox"/> Obligatorio <input type="checkbox"/> Optativo <input checked="" type="checkbox"/> Electivo					
6. Perfil de egreso-Identidad		<p>El egresado del programa de formación para la diversidad de la UC Temuco promueve prácticas inclusivas como resultado de su compromiso por el respeto a la diversidad individual, social, religiosa, de género, étnica y de capacidades puesto que enriquecen la convivencia posibilitando sociedades más justas, inclusivas, respetuosas de la diversidad.</p> <p>Este curso contribuye a ello mediante la reflexión interdisciplinar y la interacción entre estudiantes de distintas carreras.</p>					
7. Descripción		<p>El curso de Pensamiento computacional y programación para el buen vivir tiene el objetivo de acercar a los estudiantes de diversas disciplinas al mundo de la computación para favorecer espacios de diálogo que valoren la diversidad y la búsqueda de soluciones a problemas del mundo real abordados desde herramientas digitales que pueden impactar en su entorno y sus ámbitos de formación.</p> <p>En el curso habrá un acercamiento a la programación por bloque mediante Scratch usándola como medio para conocerse y compartir quiénes son y cuáles son las motivaciones que tienen para su vida personal, social y académica buscando que esto permita valorar la identidad de cada uno y fomentar la valoración por la diversidad dentro del curso y les entregue insumos para aplicar en otras experiencias de su vida.</p> <p>Además, se promoverá el aprendizaje y diseño de prototipos de aplicaciones web que busquen resolver alguna necesidad de su entorno de manera colaborativa, innovadora, con sentido crítico y que apunte a la justicia social y a la generación de oportunidades y accesibilidad de todas y todos para fomentar el buen vivir. Para esto se usarán herramientas de prototipado como Genially y</p>					





	el diseño de aplicaciones de software en su versión inicial para el sistema operativo Android con APP Inventor, que puedan dar pie a ideas innovadoras de alto impacto para sus comunidades y el desarrollo personal, social y académico.
8. Requisitos	Sin pre-requisitos
9. Competencias Específicas y nivel de dominio	No aplica
10. Competencias Genéricas y nivel de dominio	<p>Respeto y Valoración de la Diversidad: Se reconoce y reconoce a los demás en sus diferencias humanas, sean éstas socioeconómicas, etnoculturales, político-ideológicas, religioso-espirituales, de edad, de género, afectivo-sexuales, de capacidades físico-cognitivas u otras, favoreciendo un diálogo intercultural humanizador y promotor de entornos sociales más justos e inclusivos.</p> <p>Nivel de dominio 2: Reconoce el valor y la legitimidad de la diversidad propia y del otro, y comprende la riqueza de las diferencias socioculturales y de capacidades, las que incorpora en la interacción cotidiana de sus propias prácticas a fin de ir configurando su futuro desempeño profesional.</p>
11. Resultados de aprendizaje (RA)	<ul style="list-style-type: none">● Diseña una secuencia de programación para profundizar en su conocimiento personal e identitario y compartir sus características personales, intereses y motivaciones considerando aspectos individuales, sociales, culturales, religiosos, económicos, entre otros, que permitan promover el respeto y valoración por la diversidad de los estudiantes.● Crea un prototipo de aplicación web, de manera colaborativa e interdisciplinaria, para resolver alguna problemática de su entorno que se asocie con la atención a la diversidad, promoviendo el rol activo, la visión crítica y el buen vivir en la sociedad.
12. Contenidos asociados (conceptuales, procedimentales y actitudinales requeridos para el logro de cada RA)	<p>CONCEPTUALES</p> <ul style="list-style-type: none">● Pensamiento computacional● Historia de la computación, bases y evolución en el tiempo● Lenguaje de programación● Pensamiento de programación por bloques con Scratch, variables y funciones● Etapas y fundamentos del Design thinking y la importancia del prototipado● Aplicaciones web para la creación de recursos educativos digitales● App Inventor para aplicaciones en Android <p>PROCEDIMENTALES</p> <ul style="list-style-type: none">● Métodos de análisis y evaluación crítica en la búsqueda de resultados.● Uso de la información presentada en forma de hipervínculos, en forma no textual.● Socialización de contenidos digitales en múltiples dispositivos.● Uso de diversas herramientas digitales para facilitar y mejorar los procesos de colaboración.● Diseño de prototipo con Genially usando las interacciones y visualización para móvil.● Interacciones y conversaciones en diferentes situaciones y contextos.● Creación contenidos digitales para apoyar sus propias ideas y opiniones.● Utilización de herramientas y aplicaciones para mejorar la accesibilidad de los contenidos digitales.



	<ul style="list-style-type: none">• Métodos para combinar un conjunto de bloques de programa que permitan comunicar una idea y convertir su idea en una acción. <p>ACTITUDINALES</p> <ul style="list-style-type: none">• Motivación para la creación de nuevos productos y servicios con el fin de generar valor para otros.• Enfoque activo y basado en la curiosidad para explorar el funcionamiento de las tecnologías digitales.• Comportamiento ético como uno de los pilares fundamentales a la hora de desarrollar o implementar sistemas.• Actitud solidaria con los demás para mejorar sus contenidos digitales.• Apertura para explorar vías alternativas en la búsqueda de soluciones.• Cuidado por la privacidad de la información personal y ajena.• Adopción de una perspectiva empática en la comunicación siendo sensible a las emociones y experiencias de otra persona.• Confianza para lograr los objetivos del grupo.
<p>13. Estrategias de Enseñanza-Aprendizaje</p>	<p>El curso se desarrolla de acuerdo al modelo de aprendizaje experiencial para aprender Scratch y al Design Thinking con foco en el diseño de un prototipo que resuelva una problemática de su entorno.</p> <p>En ese sentido se utilizarán técnicas como:</p> <ul style="list-style-type: none">- Organizadores visuales y lluvia de ideas- Aprendizaje experiencial- Didáctica de la programación- Design Thinking- Retroalimentación para la mejora- Demostraciones públicas
<p>14. Estrategias de evaluación</p>	<p>Puesto que los estudiantes tendrán un rol central y activo, la evaluación se estructurará en torno a procesos de autoevaluación-coevaluación-heteroevaluación y espacios de retroalimentación a partir de criterios conocidos y dialogados.</p> <p>Estos procesos considerarán los siguientes momentos:</p> <ul style="list-style-type: none">- Evaluación diagnóstica para recoger conocimientos previos. Al inicio del curso.- Evaluación de proceso con foco en los conocimientos conceptuales, procedimentales y actitudinales declarados que llevarán a la presentación de algunos productos o evidencias de logro:<ul style="list-style-type: none">- Presentación del diseño de una secuencia de programación en Scratch que permita compartir: características personales, intereses y motivaciones visibilizando la diversidad individual, social, cultural, religiosa, económica, poniendo énfasis en el respetar y valorar la diversidad de los estudiantes.- Difusión de la creación de un prototipo de aplicación web en Genially de manera colaborativa, debe haber resuelto alguna problemática de su entorno incorporando una visión crítica y el buen vivir en la sociedad.- Evaluación de cierre: centrada en la valoración de las aplicaciones diseñadas y en la evaluación del proceso individual y grupal.



15. Recursos de Aprendizaje

Recursos bibliográficos

- Czerkawski, B. C., & Lyman, E. W. (2015). Exploring Issues About Computational Thinking in Higher Education. TechTrends, 59(2), 57-65. <https://doi.org/10.1007/s11528-015-0840-3>
- Gonzalo Martín Rodríguez Carrillo. (2014). Enseñanza de la programación de computadoras para principiantes: un contexto histórico. Revista Inventum, 9(17), 51-61. <https://doi.org/10.26620/uniminuto.inventum.9.17.2014.51-61>
- Laura-Ochoa, L., & Bedregal-Alpaca, N. (2021). Análisis de entornos de programación para el desarrollo de habilidades del pensamiento computacional y enseñanza de programación a principiantes. Revista Ibérica De Sistemas de Tecnologías De información, 533-548. <https://proxybiblioteca.idm.oclc.org/login?url=https://www.proquest.com/sc-holarly-journals/análisis-de-entornos-programación-para-el/docview/2562271519/se-2>
- Fundación cruzando (2019) Guía curricular Computación creativa Scratch 3.0. Ucorp, conocimiento y productividad. Chile
- Leinonen, T., & Durall, E. (2014). Pensamiento de diseño y aprendizaje colaborativo/Design Thinking and Collaborative Learning. Comunicar, 21(42), 107-116. <https://proxybiblioteca.idm.oclc.org/login?url=https://www.proquest.com/scholarly-journals/pensamiento-de-diseño-y-aprendizaje-colaborativo/docview/1476812681/se-2>
- Martínez, J. G., Minguell, M. E., & Bosch, M. P. (2018). ¿Robots o programación? El concepto de Pensamiento Computacional y los futuros maestros. [Robotics or Coding? The Concept of Computational Thinking in Pre-service Teachers] Education in the Knowledge Society, 19(2), 29-45. <https://doi.org/10.14201/eks20181922945>

Recursos informáticos

- Scratch: <https://scratch.mit.edu/>
- Guía Scratch
- Genially: <https://app.genial.ly/dashboard>

Todos los recursos estarán disponible en EDUCA.