## 1. IDENTIFICACIÓN

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Nivel de Formación[1] | Profesional | | | | | | | | | | | |
| Facultad o Unidad Académica | Facultad de Ingeniería | | | | | | | | | | | |
| Programa Académico | Ingeniería de Software | | | | | | | | | | | |
| Pensum | Profesional Específica | | | | | | | | | | | |
| Nombre de la asignatura | Introducción a la Ingeniería de Software | | | | | | | | | | | |
| Eje | Disciplinar | | | | | | | | | | | |
| Modalidad | Presencial | X | Virtual | X | Bimodal | | Dual | | | | | |
| Nivel | I | | Código de Curso | | | | | | | | | |
| Área | Básica | | Profesional Específica | X | Socio Humana | | Electiva | | | | | |
| Tipo de asignatura | Teórica _____ | | Teórico – práctico ___X__ | | | | Práctica _____ | | | | | |
| Número de créditos | 3 | | | | | | | | | | | |
| Total, Horas de Trabajo Académico | 144 | | HTD[2] | | 48 | | HTI[3] | | 96 | | | |
| Prerrequisito: | | | Correquisito: | | | | | | | | | |

## 2. JUSTIFICACIÓN:

La materia Introducción a la Ingeniería de Software es el punto de partida para generar la formación profesional de los futuros ingenieros e ingenieras de software de UNIPROPIA, en la medida en que propone

[1] Refiere a: tecnológico, profesional, especialización, maestría.

[2] Horas de trabajo Directo. HAD**.** "El trabajo directo requiere del acompañamiento del profesor, implica la presencia de estudiante (real o virtual, dependiendo de la modalidad), en actividades planificadas, que ameritan la supervisión, compañía o dirección de un profesor, e incluye acciones tendientes al cumplimiento de los objetivos del curso".

[3] Horas de Tiempo Independiente. HTI. "Comúnmente el trabajo, independiente del estudiante, debe estar relacionado con el repaso de los temas propios del contenido de los cursos, le lectura previa y posterior de la literatura sugerida en el contenido programático, la realización de actividades académicas que incluyen elaboración de ensayos, cuestionarios, relatorías, revisiones escritas de literatura especializada, consultas temáticas, resúmenes o proyectos específicos (de investigación, de diseño, de aplicación, preprácticas)".

un conocimiento integral sobre los fundamentos, principios, procesos y responsabilidades que componen la ingeniería de software, ajustando tanto las líneas técnicas contemporáneas como las perspectivas éticas, culturales y sociales que deben guiar la práctica de la ingeniería de software en los diferentes contextos.

En el marco de un proyecto educativo afrocentrado, esta materia es muy significativa puesto que permite comprender la ingeniería de software más allá de un enfoque técnico, sino como una práctica social, crítica y creativa que tiene que responder a los intereses y necesidades de las comunidades afrodescendientes de Colombia y de la diáspora africana. En el marco del estudio de los paradigmas de desarrollo, los ciclos de vida, las metodologías ágiles y los roles del ingeniero de software permite una mirada amplia donde se articula la tecnología con sus implicaciones éticas, políticas y territoriales.

Del mismo modo, esta materia introduce a los estudiantes en las transformaciones que han generado la industria del software en diversos sectores económicos y sociales, permitiendo reflexionar en torno a la soberanía digital, el racismo algorítmico, la justicia tecnológica y la pertinencia cultural en el diseño de soluciones digitales. Tal y como era el propósito, permite preparar a los estudiantes para que desde sus propias formaciones puedan tener una postura crítica y comprometida con el desarrollo tecnológico situado, responsable y participativo.

## 3. OBJETIVO GENERAL DE LA ASIGNATURA

Instruir al estudiante en los saberes básicos de la ingeniería de software, sus procesos y metodologías, y sus campos de aplicación, favoreciendo la reflexión crítica y la comprensión ética, así como la capacidad para conectar el ejercicio de la práctica profesional con necesidades tecnológicas, sociales y comunitarias de los territorios afrodescendientes.

## 4. RESULTADOS DE APRENDIZAJE Y CONTENIDO PROGRAMÁTICO

| No. | UNIDAD | CONCEPTOS BÁSICOS | EVIDENCIA DE APRENDIZAJE |
|---|---|---|---|
| 1 | Fundamentos y Contexto de la Ingeniería de Software | 1. Definición, historia y evolución de la ingeniería de software.<br>2. El software como producto y servicio.<br>3. Características y clasificación del software.<br>4. Problemas clásicos en el desarrollo de software.<br>5. Ingeniería de software en el contexto afrocolombiano: soberanía digital y equidad tecnológica. | 1. Ensayo crítico sobre el papel social y territorial de la ingeniería de software en comunidades afrocolombianas. |
| 2 | Procesos de Desarrollo y Ciclos de Vida del Software | 1. Concepto de proceso de software.<br>2. Modelos tradicionales: cascada, incremental, espiral, prototipos. | 2. Mapa conceptual del ciclo de vida del software comparando metodologías |

| | | | |
|---|---|---|---|
| | | 3. Metodologías ágiles: Scrum, XP, Kanban.<br>4. Documentación inicial del software: visión, requerimientos y diseño preliminar.<br>5. Diseño de procesos para proyectos con enfoque comunitario. | tradicionales y ágiles en contextos comunitarios. |
| 3 | Roles, Equipos y Gestión Ética en Ingeniería de Software | 1. Roles profesionales en el desarrollo de software.<br>2. Trabajo colaborativo y equipos multidisciplinarios.<br>3. Ética profesional según ACM/IEEE.<br>4. Racismo algorítmico y justicia digital.<br>5. Responsabilidad social del ingeniero de software. | 3. Análisis de caso real sobre impacto ético o social del software, presentado en formato informe. |
| 4 | Tendencias Tecnológicas y Campos de Aplicación | 1. Inteligencia artificial y aprendizaje automático.<br>2. Ciberseguridad y protección de datos.<br>3. Computación en la nube.<br>4. Analítica de datos e ingeniería de datos.<br>5. Software para la preservación cultural afrodescendiente. | 4. Propuesta de idea de proyecto tecnológico orientado a resolver una problemática en un territorio afrocolombiano. |

## 5. PLAN DE EVALUACIÓN

El plan de evaluación de la asignatura de Introducción a la Ingeniería de Software está diseñado de forma formativa/integral, entendiendo que se debe valorar la apropiación conceptual, así como la capacidad crítica, ética y contextual del estudiante para poder dar cuenta del campo de la ingeniería de software. La evaluación busca ayudar en el proceso de aprendizaje en forma progresiva a partir de actividades que permitan evidenciar competencias técnicas, analíticas y reflexivas en varias formas. Esto significa que la evaluación no se limita a evaluar resultados aislados. Lo que pretendemos es potenciar la posibilidad de que se desarrollen habilidades para la práctica, el entendimiento de las consecuencias sociales de la tecnología y la posibilidad de plantear soluciones situadas a contextos de comunidades afrodescendientes.

| Actividad | Porcentaje | Descripción |
|---|---|---|
| Talleres y ejercicios guiados | 20% | Actividades periódicas para fortalecer conceptos fundamentales. |
| Ensayo crítico (Unidad 1) | 15% | Reflexión escrita sobre el rol social de la ingeniería de software. |
| Mapa conceptual (Unidad 2) | 15% | Representación visual de los modelos de desarrollo y su pertinencia comunitaria. |
| Análisis de caso ético (Unidad 3) | 20% | Estudio individual sobre un tema ético o social |

| | | |
|---|---|---|
| | | vinculado al software. |
| Proyecto final (Unidad 4) | 30% | Propuesta de solución tecnológica contextualizada en una comunidad afro. |

## 6. BIBLIOGRAFÍA

### Bibliografía básica


- Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill. https://www.mheducation.com
- Sommerville, I. (2016). *Software engineering* (10th ed.). Pearson. https://www.pearson.com
- IEEE Computer Society & ACM. (2020). *Guide to the software engineering body of knowledge (SWEBOK V3)*. IEEE. https://www.computer.org/education/swebok
- Noble, S. U. (2018). *Algorithms of oppression: How search engines reinforce racism*. NYU Press. https://nyupress.org


### Bibliografía complementaria


- Bass, L., Clements, P., & Kazman, R. (2021). *Software architecture in practice* (4th ed.). Addison-Wesley Professional. https://www.informit.com
- Sánchez, J., & Acevedo, D. (2022). Ética en inteligencia artificial: un enfoque latinoamericano. *Revista Colombiana de Computación, 23*(1), 45–60.
- Escobar, A. (2016). *Autonomía y diseño: La realización de lo comunal*. Universidad del Cauca. https://www.unicauca.edu.co
- Comisión Económica para América Latina y el Caribe. (2022). *Inclusión digital en América Latina y el Caribe*. CEPAL. https://www.cepal.org/es/publicaciones


| CONTROL DE CAMBIOS AL SYLLABUS | | | |
|---|---|---|---|
| Elaboró / actualizó (Nombre y cargo) | | Revisó (Nombre y cargo) | |
| Fecha de aprobación del Comité de Currículo: | | Acta de aprobación del Comité de Currículo: | |
| Cambios realizados | | | |