



Fecha: junio 2021

PROGRAMA ACADÉMICO: INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

SEMESTRE: TERCERO

ASIGNATURA: PROGRAMACIÓN II

CÓDIGO: 8104773

NÚMERO DE CRÉDITOS: 4

PRESENTACIÓN

El objetivo de esta asignatura es que el estudiante al finalizar el curso tenga la capacidad de construir una aplicación completa, es decir con varias interfaces de usuario, con persistencia (usando diferentes formatos de archivos) y con una arquitectura que permita obtener un producto eficiente, escalable, robusto, mantenible, usable, funcional y en términos general con calidad.

Durante el curso el estudiante conocerá diferentes estilos, arquitecturas, tecnologías y en general herramientas que permiten complementar sus conocimientos en programación, por ejemplo, a nivel de interfaz de usuario se repasaran los principios básicos y los diferentes estilos actuales del área (móvil, web y de escritorio entre otros).

Al terminar el curso el estudiante habrá mejorado sus habilidades de codificación y de trabajo en equipo, por ello se hará énfasis en buenas y mejores prácticas de programación, un estilo organizado, en la documentación del código, en la constante revisión y mejora de sus aportes de código a un proyecto y en otras habilidades requeridas por un programador para producir código de calidad.

JUSTIFICACIÓN

En el mundo laboral, se requieren profesionales capaces de afrontar situaciones reales y dinámicas, que estén focalizados en la gestión y desarrollo de las mejoras continuas. Por ello esta asignatura contribuye a formar las bases fundamentales del futuro Ingeniero de Sistemas y Computación, que esté en la capacidad de aplicar los conocimientos adquiridos en la construcción de soluciones a los problemas presentados, haciendo uso de diferentes herramientas computacionales.

Luego de analizar el ámbito internacional es evidente que una de las habilidades más demandadas por las compañías de Tecnología Informática, es la de desarrollar aplicaciones, según la "Encuesta de Salarios IT 2017", realizada por el Portal Computer World; en el campo nacional el panorama es similar teniendo en cuenta el "Estudio de salarios y profesionales del sector de Software y TI de Colombia 2016", elaborado por MinTIC y FEDESOFTE, donde se presentan los perfiles más solicitados en la industria nacional, destacándose los relacionados con el área de desarrollo, lo que es corroborado con otros estudios que señalan el déficit de profesionales con estos perfiles, por ello la estructura de esta asignatura contribuye de manera efectiva a responder a las necesidades de esta industria.

Por otra parte, es justo mencionar que las habilidades que se desarrollan en esta asignatura sirven como base para el abordaje de asignaturas de otras áreas que hacen parte del plan de estudios del programa.

Por último, es importante resaltar que la lógica de programación actualmente se considera una habilidad transversal, debido a que contribuye al desarrollo del pensamiento lógico y matemático que se requiere en cualquier campo de la ingeniería. Existen además, comunidades académicas que han identificado la necesidad de incluir temas de lógica de programación desde tempranas edades en los primeros niveles de escolaridad.



COMPETENCIAS

Adquirir una visión global de la Ingeniería de sistemas en el campo laboral y construir modelos de solución algorítmicas focalizadas en las bases de la lógica y la programación, con el fin de construir soluciones reales, viables y de eficaz implementación en un lenguaje de programación

- Conoce y diferencia los formatos usados en el almacenamiento, representación, intercambio de la información
- Entiende los conceptos básicos de la comunicación hombre máquina y los aplica en la solución de problemas de interacción de software con usuarios finales
- Entiende y aplica conceptos de principios básicos de computación gráfica
- Crea y evalúa interfaces de usuario personalizables aplicando los principios de diseño de diferentes plataformas
- Construye componentes gráficos reutilizables
- Utiliza servicios Web
- Da soluciones a problemas por medio de aplicaciones funcionales completas
- Utiliza herramientas de software para soportar los procesos de las etapas del ciclo de vida del desarrollo de software
- Trabaja en equipo para dar solución a problemas planteados
- Genera código de calidad aplicando buenas prácticas de programación

RESULTADOS DE APRENDIZAJE

- Utilizo diferentes formatos (archivos planos, binarios, XML, JSON) usados para el almacenamiento, representación e intercambio de la información (consumo de servicios web), como parte de las soluciones de software dando respuesta a las necesidades de diferentes contextos organizacionales.
- Construyo interfaces gráficas personalizables y componentes gráficos reutilizables como parte de aplicaciones funcionales completas, aplicando los conceptos básicos de la comunicación hombre máquina en la solución de problemas de interacción de software con usuarios finales, a partir del trabajo en equipo y aplicando buenas prácticas en la construcción de software.
- Empleo herramientas especializadas en cada una de las etapas del ciclo de vida de desarrollo de software.
- Leo documentos técnicos en idioma inglés y utilizo términos propios del mismo en el proceso de codificación.

METODOLOGÍA

Esta asignatura será guiada en los espacios presenciales por la complementación conceptual del docente al trabajo de preparación previo que los estudiantes han realizado sobre la temática particular a tratar en la sesión; por lo tanto, un tema será abarcado en cuatro momentos:

1. Preparación, consulta e investigación conceptual por cuenta del estudiante y su pequeño grupo de trabajo.
2. Tratamiento conceptual del tema en sesión del gran grupo junto con el docente.
3. Aplicación de talleres individuales y cooperativos a nivel tutorial.
4. Desarrollo de actividades de refuerzo en sesiones autónomas.
5. Se tendrá en cuenta el aprendizaje basado en proyectos

De lo anterior se verifica que en la actividad 1, el estudiante constituirá conflictos conceptuales de baja



complejidad, a solucionar en el transcurso de la actividad 2, entre tanto, la actividad 4 generará conflictos cognitivos orientados a la aplicación, a subsanar con la actividad 3.

Se considera que el estudiante debe alcanzar un amplio trabajo autónomo, que posteriormente será complementado por el trabajo cooperativo de su pequeño grupo. La generación de conflictos cognitivos es importante y necesaria, para que las sesiones de gran grupo cumplan con su objetivo de afianzamiento del conocimiento.

INVESTIGACIÓN

La asignatura por sus condiciones particulares, promueve en el estudiante la investigación formativa, de modo sea posible obtener los fundamentos con los que podrá posteriormente vincularse en el tratamiento de áreas específicas de la disciplina Informática, por medio de los grupos y semilleros de investigación dinamizados al interior de la Escuela de Ingeniería de Sistemas y Computación.

MEDIOS AUDIOVISUALES

Salas de informática

La asignatura es eminentemente práctica, de manera que se hace necesario que todas sus clases sean cumplidas en salas de informática.

EVALUACIÓN

EVALUACIÓN COLECTIVA

No aplica.

EVALUACIÓN INDIVIDUAL

La evaluación en la asignatura está orientada a determinar el nivel de desarrollo de los procesos lógicos en el estudiante, junto con su capacidad para abstraer problemas y generar soluciones informáticas.

RANGOS DE VALORACIÓN (%)

El cálculo de la nota final se hará de la siguiente manera:

- 70% Evaluaciones
- 15% Talleres y exposiciones
- 15% Trabajos

CONTENIDOS TEMÁTICOS CENTRALES

1. PERSISTENCIA

- 1.1. Archivo de propiedades (multilenguaje)
- 1.2. Archivo tipo texto
- 1.3. Archivos binarios
- 1.4. Archivos serializados
- 1.5. Archivos XML
- 1.6. Archivos JSON
- 1.7. Consumo de servicios Web

2. INTERFAZ DE USUARIO Y EXPERIENCIA DE USUARIO

- 2.1. Principios para diseño de interfaz de usuario móvil
- 2.2. Principios para diseño de interfaz de usuario Web
- 2.3. Principios para diseño de interfaz de usuario de escritorio
- 2.4. Principios para diseño de interfaz de usuario por consola
- 2.5. Principios para diseño de reportes



3. PRESENTACIÓN

- 3.1. Diseño de interfaces gráficas
- 3.2. Componentes gráficos de Java (AWT, Swing, 2D, 3D)
- 3.3. Utilización de layouts
- 3.4. Manejo de eventos
- 3.5. Cambio de interfaces parametrizables

4. INTERACCION CON EL USUARIO USANDO PATRON MVP

- 4.1. Interfaces: compromisos funcionales
- 4.2. Herencia como mecanismo de reutilización
- 4.3. Arquitectura de multicapas

5. ARTEFACTOS Y ENTREGABLES

- 5.1. Generación de ejecutables
- 5.2. Entrega de proyectos

LECTURAS MÍNIMAS

Artículos y apartados bibliográficos de los diferentes temas a tratar, proporcionados por el docente. Consulta permanente a los foros de las comunidades de desarrolladores.

BIBLIOGRAFÍA

- Krug Steve, Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability (3rd Edition) (Voices That Matter), New Riders, 2014.
- Morrelli Ralph y Walde Ralph, Java, Java, Java Object-Oriented Problem Solving, Prentice Hall, 2006.
- Weinschenk Susan, Neuro Web Design: What Makes Them Click?, Peachpit Press, 2009.
- Beard Jason, The Principles of Beautiful Web Design SitePoint Pty. Limited, 2014.
- DEITEL, Harvey M; DEITEL Paul J. Cómo programar en C/C++ y Java. Pearson. 1113 p., 2004.
- Sznajdleder, Pablo. Java a fondo: estudio del lenguaje y desarrollo de aplicaciones. Alfaomega. 2013
- Wu, Introducción a la programación orientada a objetos con Java. McGraw-Hill. 2001
- Martín, Antonio. Programador certificado Java 2 curso práctico. Alfaomega. 2008.
- Muñoz, Jaime et al. Temas de diseño en interacción humano-computadora. 2014
- Montero, Yusef Hasam. Experiencia de usuario, principios y métodos.
- Yusef Hassan, Francisco J. Martín Fernández y Ghzala Iazza. Diseño Web Centrado en el Usuario: Usabilidad y Arquitectura de la Información.

En línea:

- Deitel, P. (2018). Java how to program. (11a. ed.) Pearson Educación. Tomado de <https://biblio.uptc.edu.co:2566/?il=7164>
- Blasco, F (2019). Programación Orientada a Objetos en Java. Ra-Ma. Tomado de <https://biblio.uptc.edu.co:3040/library/publication/programacion-orientada-a-objetos-en-java-1591713437>
- Flórez, Héctor (2012). Programación orientada a objetos usando Java. Ecoe Ediciones. Tomado de <https://biblio.uptc.edu.co:2540/a/30030>
- Reges, S. (2018). Building Java Programs. (4a. ed.) Pearson Educación. Tomado de <https://biblio.uptc.edu.co:2566/?il=6292>
- Liang, D. (2018). Introduction to Java Programming. (11a. ed.) Pearson Educación. Tomado de <https://biblio.uptc.edu.co:2566/?il=9153>
- Vásquez, B. (2018). Java y C++. Ediciones de la U. Tomado de <https://biblio.uptc.edu.co:2566/?il=9083>
- Ferrer, J. (2016). Desarrollo de interfaces. Ediciones de la U. Tomado de



MACROPROCESO: DOCENCIA
PROCESO: GESTIÓN DE PROGRAMAS ACADÉMICOS
PROCEDIMIENTO: FORMULACION O ACTUALIZACION DEL PROYECTO ACADEMICO EDUCATIVO-PAE PARA PROGRAMAS DE
PREGRADO
CONTENIDOS PROGRAMATICOS PROGRAMAS DE PREGRADO

Código: D-GPA-P01-F02

Versión: 02

Página 5 de 3

<https://biblio.uptc.edu.co:2566/?il=7944>

- Shneiderman, B. (2018). Designing the user interface strategies for effective human-computer interaction. (6a. ed.) Pearson Educación. Tomado de <https://biblio.uptc.edu.co:2566/?il=6373>
- Serna, S. (2017). Diseño de interfaces en aplicaciones móviles. Ediciones de la U. Tomado de <https://biblio.uptc.edu.co:2566/?il=7989>
- Cassá, O et al (2012). Programación Web en Java. Ministerio de Educación, Cultura y Deporte de España. Tomado de <https://biblio.uptc.edu.co:2540/a/20928>
- Vásquez, B. (2018). Java y C++. Ediciones de la U. Tomado de <https://biblio.uptc.edu.co:2566/?il=9083>
- Java, API especification Oracle. <https://docs.oracle.com/javase/8/docs/api/overview-summary.html>