***Universidad Nacional***

 ***Sede Regional Brunca, Campus Pérez Zeledón***

***Facultad de Ciencias Exactas y Naturales***

***Escuela de Informática***

**PROGRAMA DE CURSO**

CÓDIGO: EIF209

NRC: 43432

NOMBRE: Programación IV

CARRERA: Ingeniería en Sistemas de Información

REQUISITOS: EIF206 Programación III

TIPO DE CURSO: Común

ÁREA DISCIPLINARIA: Ingeniería de software

NATURALEZA: Teórico / Práctico

MODALIDAD: Ciclo 16 semanas, modalidad presencial

CICLO LECTIVO: I ciclo 2023

NIVEL: III nivel (BA-INFORM)

GRUPO: 87

PROFESOR: Ruben Mora Vargas

CORREOS: [rubenmoravargas@gmail.com](mailto:rubenmoravargas@gmail.com)

HORARIO: lunes 17:00 a 21:00

HORARIO ATENCIÓN: lunes 21:00 a 22:00

GUÍA ACADÉMICO: Máster Alejandro Flores Quesada.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Créditos | Horas Semanales | Horas Presenciales | | Horas estudio independiente |
| 4 | 11 | Teóricas | Prácticas | 7 |
| 2 | 2 |

***En esta Universidad nos comprometemos a prevenir, investigar y sancionar el hostigamiento sexual entendido como toda conducta o comportamiento físico, verbal, no verbal escrito, expreso o implícito, de connotación sexual, no deseado o molesto para quien o quienes lo reciben, reiterado o aislado. Si usted está siendo víctima de hostigamiento diríjase a la Fiscalía de Hostigamiento Sexual de la UNA o llame al teléfono: 2277-3961.***

**DESCRIPCIÓN**

Este curso trata varios temas relacionados con el desarrollo de aplicaciones web que operan en Internet. Se construyen aplicaciones de gestión estructuradas por capas y con interfaz web. En la parte del cliente, se integran los conceptos del lenguaje de marcas HTML con páginas dinámicas utilizando lenguajes de scripting y hojas de estilo. En la parte servidor se integra un lenguaje orientado a objetos, un modelo de aplicación, conexión a bases de datos y plataformas de trabajo basadas en componentes (frameworks). La práctica consiste en proyectos que los estudiantes irán desarrollando a lo largo del curso.

**OBJETIVO GENERAL**

Que el estudiante sea capaz de desarrollar aplicaciones web que operen en Internet, integrando diversas técnicas y tecnologías en la implementación tanto de la parte cliente como de la parte del servidor y aplicando principios adecuados de ingeniería de software.

**OBJETIVOS ESPECÍFICOS**

1. Comprender el modo de operación de Internet y de la web.
2. Describir la estructura de una página utilizando HTML y HTML 5.
3. Trabajar el modelo de objetos (DOM) de JavaScript y las API de HTML5.
4. Utilizar XML, JSON para la descripción y serialización de datos.
5. Comprender y utilizar el modelo de páginas dinámicas por medio de servlets y JSP.
6. Desarrollar aplicaciones web por capas con conexión a bases de datos relacionales.
7. Utilizar AJAX para la construcción de páginas de contenido rico (Rich Content Pages).
8. Usar plataformas basadas en componentes para el desarrollo de aplicaciones web en Java.

**HABILIDADES Y COMPETENCIAS:** Al terminar el curso, el estudiante deberá tener las habilidades y competencias necesarias para:

1. Diseñar y construir aplicaciones web.
2. Organizar adecuadamente aplicaciones web según prácticas comunes de diseño.
3. Comprender y utilizar las diferentes técnicas y herramientas disponibles para la construcción de sitios web funcionales y seguros.
4. Trabajar adecuadamente en equipos de trabajo, asumiendo sus roles y responsabilidades y resolviendo adecuadamente los conflictos.
5. Evaluar su proceso de aprendizaje y el de sus compañeros de grupo.

**CONTENIDO**

1. **Conceptos de programación en Internet y web**
   1. Internet y web
   2. Protocolos TCP/IP y HTTP
   3. Servidor web, sitio web y página web
   4. El servidor web
   5. Las aplicaciones web
2. **Introducción al lenguaje de marcas HTML**
   1. Principales marcas
   2. Enlaces
   3. Formularios
   4. Hojas de estilo en cascada (CSS3)
   5. Fundamentos de HTML5
3. **Introducción a la programación con scripts**
   1. El lenguaje JavaScript
   2. Estructuras de datos y de control
   3. Funciones definidas por el usuario y de biblioteca
   4. El modelo DOM y el procesamiento de eventos
   5. Expresiones regulares
   6. Errores comunes
   7. La API de HTML5
      1. Herramientas gráficas y capacidades multimediales
      2. Verificación de formularios y bases de datos locales
      3. Geolocalización
4. **El modelo de capas**
   1. Arquitectura del modelo de capas
   2. Capa de presentación
   3. Capa de la lógica de aplicación
   4. Capa de acceso a la base de datos
   5. Modelo conceptual y modelo de clases en UML
5. **Capa de presentación con páginas dinámicas**
   1. Páginas dinámicas
   2. Procesamiento de peticiones (request)
   3. Generación de respuestas (response)
   4. Servlets yJava Server Pages (JSP)
   5. Conceptos básicos de AJAX
6. **Capa lógica de aplicación**
   1. Objetos, clases e interfaces
   2. Asociaciones y herencias
   3. Gestores
7. **Capa de acceso a la base de datos**
   1. Modelo de objetos versus modelo de datos
   2. Conexión a bases de datos
   3. Objetos de datos: conexiones, sentencias o comandos, resultados
8. **Validación en el cliente y en el servidor**
   1. Validación mediante scripts en el cliente
   2. Validación en el servidor
9. **Administración de sesiones**
   1. Definición y seguimiento de sesiones
   2. Justificación del uso de sesiones
10. **Uso de frameworks**
    1. El patrón de arquitectura MVC (Model-View-Controller)
    2. Las fases del ciclo de procesamiento
    3. Ligado de componentes (backing beans binding)
    4. Ligado de propiedades (value binding)
    5. Biblioteca de componentes
11. **Autenticación y seguridad**
    1. Protección de datos en la web
    2. Autenticación en el protocolo http
    3. Autenticación usando la base de datos
12. **Diseño de aplicaciones web**
    1. Servicios web

**METODOLOGÍA**

Las aplicaciones web, al contrario de las aplicaciones locales, se componen al menos de dos partes, que operan en distintas computadoras interconectadas por la red Internet. La implementación tanto de la parte cliente como de la parte servidor de la aplicación requiere no solo conocer las tecnologías que se usan en cada parte sino los mecanismos para su adecuada integración.

La estrategia seguida en este curso está orientada a la modalidad presencial, y divide su temática en cuatro grandes secciones con énfasis práctico. Primero se ubica al estudiante en un marco general de cómo operan las aplicaciones web. Luego, se estudian las tecnologías y lenguajes requeridos en la parte cliente de la aplicación. Posteriormente se hace lo propio con la parte servidor, para tener así una aplicación completa. En la última parte del curso se estudia y aplica alguna plataforma de componentes (framework) que permita una mayor productividad en el desarrollo de este tipo de aplicaciones. El curso consta de actividades y laboratorios prácticos en los cuales el estudiante aplicará los conceptos expuestos.

También, habrá un proyecto más complejo, en donde el estudiante deberá integrar los distintos conceptos al crear aplicaciones web, bajo el asesoramiento del profesor.

El curso consta de clases magistrales impartidas por el profesor, seguidas por sesiones prácticas o pequeños laboratorios para que el estudiante aplique los conceptos expuestos.

También habrá proyectos extraclase más complejos donde el estudiante deberá integrar los distintos conceptos al crear aplicaciones de software para web, por supuesto con el asesoramiento del profesor.

También se invitará a expertos en algunos temas para complementar y motivar el aprendizaje.

El curso propicia el desarrollo de destrezas de diseño de software, enfrentando a los estudiantes a proyectos de mediana complejidad que ellos deberán completar de principio a fin. También se refuerza la habilidad de la reutilización de software, pues tanto en las sesiones de laboratorio como en los exámenes y proyectos los estudiantes tendrán a disposición todos los ejemplos y prácticas anteriores, de manera qué rápidamente puedan adaptar una solución existente a una nueva necesidad más o menos semejante.

La asistencia oportuna y comprometida del estudiante a las horas de consulta le permitirá obtener del profesor en este espacio: orientación en los trabajos asignados durante el curso, evacuación de dudas en los temas estudiados y la articulación conjunta de ideas para el desarrollo de los trabajos. Este horario no descarta la posibilidad de que los estudiantes planteen dudas y soliciten orientación por otros medios, como el correo electrónico o plataforma utilizada.

El acceso oficial a la documentación, material y entrega de asignaciones del curso, al igual que los avisos/mensajes oficiales (contacto/correos especificados en la plataforma) estarán centralizados mediante el uso del sistema de Aula Virtual UNA (https://www.aulavirtual.una.ac.cr/).

**MEDIACIÓN PEDAGÓGICA:**

A continuación, se describe el formato base general del protocolo a seguir en la realización de los laboratorios prácticos y la estructura básica en la evaluación del proyecto programado, actividades que constituyen la base de la evaluación de las competencias en el curso, **todas las evaluaciones serán presenciales**.

**EVALUACIÓN**

|  |  |
| --- | --- |
| **Descripción** | **Porcentaje** |
| **Examen parcial**  Cubre los objetivos del curso hasta los temas de la semana 6. | 20% |
| **Examen final**  Cubre todos los objetivos del curso. | 20% |
| **Proyecto del Curso**  Sirve para evaluar aspectos prácticos concretos de los temas estudiados en el curso. Deben realizarse en grupos de tres personas exactas, según lo indique el profesor en el momento que se entregue el enunciado de este. Se trabaja con 4 etapas:  Base Inicial (8%): Se revisa la arquitectura y diseño del trabajo, diseño de datos y páginas web y la conexión y diseño con el API.  Aplicación Base (10%): Se revisan correcciones anteriores, las funcionalidades del sistema final y se proponen cambios para la defensa final, refactorización de código, y seguridad.  Defensa del proyecto (14%): Se revisan correcciones anteriores, la implementación del sistema, pruebas de estrés de la aplicación con datos, Cumplimiento de requerimientos y pruebas de integración.  **Nota: Si el proyecto no tiene un 80% de la completitud solicitada pierde la posibilidad de realizar la presentación final y el puntaje respectivo.**    Presentación Final (8%): Se revisan correcciones anteriores, y detalles que el profesor indica, realiza una presentación a los compañeros, profesor e invitados, presenta alcance, conclusiones y recomendaciones del proyecto. | 40% |
| **Evaluaciones cortas**: Se realizan dos evaluaciones según cronograma | 20% |
| **Total** | **100%** |

**CONSIDERACIONES GENERALES:**

1. Se pueden mover contenidos de su respectiva semana cuando se considere conveniente sin que se incurra en perjuicio del estudiante.
2. No se aceptarán trabajos fuera de las fechas y las horas establecidas. Queda a discreción del profesor aceptarlos, pero si fuese así, con una penalidad por el retraso. Queda a criterio del profesor el valor de esa penalización en forma general o caso por caso.
3. Al ser un curso cuya evaluación contempla aspectos que se desarrollan a lo largo del curso, como proyectos programados, el curso **no tiene examen extraordinario**.
4. Como directriz de la cátedra se establece como obligatoria la defensa oral de los proyectos, por consiguiente, se deberá defender de manera presencial y en caso de cambios en la modalidad se deberá elaborar un video bien producido, demostrando el funcionamiento del proyecto desarrollado, si este no es el caso el trabajo **no** será evaluado.
5. Todo código entregado en una evaluación debe ser por medio de un repositorio github privado compartido con el usuario **rubenmoravargas**.
6. Todo código debe estar en inglés, seguir las buenas prácticas que se establecen en clase.
7. Todo documento escrito debe tener su respectiva portada, índice, bibliografía y conclusiones.
8. Los trabajos digitales deben ser subidos a la plataforma Aula Virtual, en las fechas y horas establecidas por el profesor. Generalmente será un archivo de texto con el código el link del repositorio y/o la carpeta de drive donde está la documentación. No se permite enviar los trabajos al correo del profesor a menos que sea una situación especial y el profesor lo haya autorizado. No subir los archivos del trabajo dejan sin puntaje y sin la posibilidad de reclamos.
9. Los archivos para subir deben estar debidamente identificados con todos los nombres de los integrantes del grupo. Con el siguiente formato:
   1. Se debe indicar la asignación en mayúsculas, seguido de un guión bajo “\_”. Ejemplos: **PROYECTO\_** o **LAB1\_**.
   2. Luego, indicar por medio de la inicial del nombre seguido del apellido del estudiante (cuando existan varios integrantes se deben separar por medio de un guión bajo “\_”). Todo en minúsculas, sin signos de puntuación y acentuación. Ejemplo: si los estudiantes son Nikola Tesla y Thomas Alva Edison y entregarán el primer proyecto, entonces el nombre de la carpeta será **PROYECTO\_ntesla\_tedison**.
   3. Antes de subir los archivos al Aula Virtual, estos deben ir comprimidos con el formato **ZIP**.
   4. Para documentos, estos deben estar en formato docx para poder calificarlos.
   5. No se permite incluir, luego de enviado el trabajo, a ningún estudiante.
   6. Si un estudiante incumple el formato de entrega, se le penalizará, esa penalización quedará a criterio del profesor. Si es un trabajo en grupo se penalizará a todo el grupo.
10. Todos los derechos de autor están reservados, por consiguiente, se prohíbe la reproducción parcial o total del material provisto en cada lección. Cada lección en línea y su correspondiente material (grabaciones, presentaciones y documentos) están protegidos por las leyes de derechos de autor, las penales y civiles. Por lo tanto, el profesor se reserva los derechos de reproducción del material suministrado. Igualmente, se prohíbe utilizar la imagen y el sonido del curso en línea para fines no académicos o transferirlos a terceros. La reproducción de cualquier recurso escrito, visual o auditivo sin los correspondientes permisos será sancionada según lo establecen las leyes civiles, la Ley de Derecho de Autor, la Ley de Procedimientos de Observancia y el Código Penal, en caso de incurrir en alguna infracción sin perjuicio de las sanciones disciplinarias internas de la Universidad Nacional.
11. Si se determina que existe plagio de proyectos o tareas, o copia en los exámenes se procederá a establecer como calificación respectiva un 0 a todos los involucrados y el caso se comunicará a las autoridades respectivas de la universidad para que se tomen las medidas necesarias. El plagio y la copia no se permiten y serán sancionados según lo establecen los reglamentos de la institución. Ver artículos 24 y 25 del Reglamento General sobre los Procesos de Enseñanza Aprendizaje.

**Reglamento general sobre los Procesos de Enseñanza y aprendizaje de la Universidad Nacional**

**Artículo 24. Plagio –** *Se considera plagio la reproducción parcial o total de documentos ajenos presentándose como propios. En el caso que se compruebe el plagio por parte del estudiante, perderá el curso. Si reincide será suspendido de la carrera por un ciclo lectivo, y si la situación se repite una vez más, será expulsado de la Universidad.*

**Artículo 25. Copia –** *Se considera copia todo documento o medio no autorizado utilizado de manera subrepticia por el estudiante durante una prueba evaluativa. De comprobarse la copia en la realización de una evaluación, ésta será calificada con nota de cero y el estudiante perderá el porcentaje correspondiente a esa evaluación, independientemente de la eventual sanción disciplinaria establecida en la normativa institucional.*

**CRONOGRAMA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SESIÓN** | **FECHA** | **TIPO DE SESIÓN** | **APRENDIZAJES INTEGRALES** | **ACTIVIDADES** | **RECURSOS DIDÁCTICOS** |
| **1** | 27 FEB | Presencial | **Entrega de programa de curso**  Tema II | Desarrollo del tema | Diapositivas, Apps, Aula Virtual, Telegram |
| **2** | 6 MAR | Presencial | Tema II  **Asignación Proyecto** | Desarrollo del tema | Computadora, Apps, Aula Virtual, Telegram |
| **3** | 13 MAR | Presencial | Tema I  **Evaluación I** | Desarrollo del tema  **Evaluación** | Computadora, Apps, Aula Virtual, Telegram |
| **4** | 20 MAR | Presencial | Tema III | Desarrollo del tema | Computadora, Apps, Aula Virtual, Telegram |
| **5** | 27 MAR | Presencial | Tema III  **Evaluación II** | Desarrollo del tema  **Evaluación** | Computadora, Apps, Aula Virtual, Telegram |
|  | ***03 – 07 ABR*** | | ***Semana Santa*** | | |
| **6** | 10 ABR | Presencial | Tema IV | Desarrollo del tema | Computadora, Apps, Aula Virtual, Telegram |
| **7** | 17 ABR | Presencial | **Examen Parcial** | **Examen** | Computadora, Apps, Aula Virtual, Telegram |
| **8** | 24 ABR | Presencial | Tema V | Desarrollo del tema | Computadora, Apps, Aula Virtual, Telegram |
| **9** | 1 MAY | Presencial | Tema V | Desarrollo del tema | Computadora, Apps, Aula Virtual, Telegram |
| **10** | 8 MAY | Presencial | Tema V  **Semana Académica** | Desarrollo del tema | Computadora, Apps, Aula Virtual, Telegram |
| **11** | 15 MAY | Presencial | Tema VI  **Entrega base Inicial** | Desarrollo del tema **Entrega de proyecto** | Computadora, Apps, Aula Virtual, Telegram |
| **12** | 22 MAY | Presencial | Tema VI | Desarrollo del tema | Computadora, Apps, Aula Virtual, Telegram |
| **13** | 29 MAY | Presencial | Tema VII  **Entrega Aplicación base** | Desarrollo del tema **Entrega de proyecto** | Computadora, Apps, Aula Virtual, Telegram |
| **14** | 5 JUN | Presencial | **Defensa del proyecto** | **Entrega de proyecto** | Computadora, Apps, Aula Virtual, Telegram |
| **15** | **12 JUN** | **Presencial** | **Examen Final** | **Examen** | Computadora, Apps, Aula Virtual, Telegram |
| **16** | 19 JUN | Presencial | **Presentación del proyecto** | **Entrega de proyecto** | Computadora, Apps, Aula Virtual, Telegram |
| **17** | **26 JUN** | **Asíncrona** | **Entrega de calificaciones finales** | **Entrega de calificaciones** | **Aula Virtual** |

**BIBLIOGRAFÍA**

* El Gran Libro de Diseño Web - Edicion Única (2012).
* Firtman, Maximiliano R.Ajax.Web2.0 para profesionales.
  + 1era edición. Alfaomega Grupo Editor S.A. de C.V. México, 2008
* Flanagan, David.JavaScript: The Definitive Guide.
  + 5th edition,O'Reilly,2006.
* Hall,Marty;Brown,Larry;Chaikin,Yaakov.CoreServletsandJavaServerPages.Volume2:AdvancedTechnologies.
  + 2nd edition, Prentice Hall, 2007.
* Hall, Marty; Brown,Larry.Core Servlets and Java ServerPages.
  + 2nd edition, Prentice Hall, 2004
* Hunter Jason; Crawford, William.Java ServletProgramming.
  + 2da Edición. O´Reilly. 2003.
* Lubbers,Peter;Albers,BrianandSalim,Frank.ProHTML5Programming:PowerfulAPIsforRicherInternet ApplicationDevelopment.
  + APress, Springer Science+Business Media, LLC. New York, USA, 2010.
* Stefanov,Stoyan; Chetan Sharma,Kumar.Object-Oriented JavaScript. 2nd edition, Packt Publishing. Birmingham B3 2PB, UK.2013

NOTA: las referencias indicadas son solamente una selección de materiales que podrían ser útiles. No hay ningún libro de texto para el curso.

+ Además de los textos anteriores, se utilizarán manuales, herramientas, documentos, revistas y bibliografía o material adicional que se anotará oportunamente durante el curso. Para la elaboración de los proyectos se utilizará software provisto de manera oportuna por el profesor.

**Bases de datos de consulta del Sistema SIDUNA.**

* **ACM**: Association for Computing Machinery.
* **EBSCO**: Computers & Applied Sciences Complete.