

FACULTAD/ESCUELA DE INGENIERÍA, DISEÑO Y CIENCIAS APLICADAS

DEPARTAMENTO DE COMPUTACIÓN Y SISTEMAS INTELIGENTES

Información de la asignatura

Nombre de la asignatura	Proyecto integrador I
Código de la asignatura	09793 - TIC
Periodo Académico	202410
Nrc	11654
Grupo	001
Programas/Semestres	TEL 05
Intensidad horaria	2
Intensidad Semanal	4
Créditos	3
Docente(s)	Domiciano Rincon Niño

Introducción o presentación general del curso

En este curso los estudiantes deben integrar y aplicar los conceptos aprendidos en los cursos de los bloques de Fundamentos de Algoritmos y Programación e Ingeniería de TI. Para esto, los estudiantes trabajan en equipo desarrollando una solución distribuida en red con datos persistentes, utilizando un modelo de ciclo de vida para el desarrollo del sistema, que ofrezca respuesta a un problema de Ingeniería Telemática de mediana complejidad, comunicando apropiadamente los resultados de cada una de las etapas del proceso.

Formación en competencias

SO-1: Capacidad para identificar, formular y resolver problemas complejos de ingeniería, mediante la aplicación de principios de ingeniería, ciencias y matemáticas.

SO-2: Capacidad para aplicar el método de diseño en ingeniería, con el fin de construir soluciones que satisfagan necesidades específicas, teniendo en cuenta aspectos de salud pública, seguridad y bienestar, así como factores globales.

SO-3: Comunicarse efectivamente de forma oral y escrita, tanto en español como en inglés

SO-5: Capacidad para trabajar de manera efectiva en un equipo, cuyos miembros impartan liderazgo, creen un ambiente de trabajo colaborativo e inclusivo, establezcan metas, planeen tareas y logren objetivos.

Objetivo general de aprendizaje

Integrar los conceptos y habilidades adquiridos en los cursos disciplinares previos, para el desarrollo en equipo de una solución distribuida en red con datos persistentes, utilizando un modelo de ciclo de vida para el desarrollo del sistema, que ofrezca respuesta a un problema de Ingeniería Telemática de mediana complejidad, comunicando apropiadamente los resultados de cada una de las etapas del proceso.

Objetivos terminales - Resultados de aprendizajes

Resultado de aprendizaje del curso o asignatura	Competencia en formación	Resultado de aprendizaje de la competencia de egreso al que se contribuye
Analizar los principales componentes para la construcción y operación de un sistema web,	SO1 - Solución de problemas	SO1-PI1 Identificación de problemas (A), SO1-PI2 Formulación de problemas (A)
Diseñar un sistema fullstack de mediana complejidad, a partir del análisis de un problema	SO1 - Solución de problemas, SO2 – Diseño en ingeniería	SO2-PI2 Selección de diseño (T), SO2-PI3 Evaluación del diseño (T)
Desarrollar un ecosistema de aplicaciones para usuario, siguiendo un modelo de desarrollo ágil	SO1 - Solución de problemas, SO5 – Trabajo en equipo	SO1-PI3 Solución de problemas (T), SO5-PI1 Formación de equipos efectivos (A), SO5-PI2 Funcionamiento del equipo (Metas, tareas, actividades) (A) SO5-PI3 Funcionamiento del equipo (liderazgo colaborativo) (A)
Participar en un equipo de ingeniería aplicando habilidades de trabajo en equipo	SO5 – Trabajo en equipo	SO5-PI1 Formación de equipos efectivos (A), SO5-PI2 Funcionamiento del equipo (Metas, tareas, actividades) (A), SO5-PI3 Funcionamiento del equipo (liderazgo colaborativo) (A)
Comunicar apropiadamente los resultados de cada una de las etapas del proceso	SO3 - Comunicación efectiva	SO3-PI1 Comunicación Escrita (A) ,SO3-PI2 Presentación Oral (T)

Unidades de aprendizaje

Unidad 1 – Gestión ágil de proyectos

- Workflow: Métodos y procesos utilizados para gestionar proyectos de manera ágil.
- Estrategias de branching: Técnicas utilizadas para organizar y gestionar el trabajo en ramas dentro de un repositorio de código.
- Product backlog: Lista priorizada de funcionalidades o características del producto que se deben desarrollar.
- Gestión del avance del proyecto: Métodos y técnicas para monitorear y controlar el progreso de un proyecto.
- Análisis PESTEL: Método de análisis que evalúa factores políticos, económicos, sociales, tecnológicos, ambientales y legales para entender el entorno en el que se desarrolla un proyecto.

Unidad 2 – Servicios en red

- Modelo OSI: Modelo de referencia que describe cómo se comunican los sistemas de redes en diferentes capas.
- Protocolos IP: Protocolos de Internet utilizados para el envío y recepción de datos en redes IP.

- Protocolos de nivel de transporte: TCP y UDP: Protocolos que permiten el transporte de datos entre aplicaciones en redes de computadoras.
- Protocolos de nivel de red: Protocolos utilizados para el enrutamiento de datos en redes de computadoras.
- ¿Qué es el backend?: Concepto que se refiere a la parte de un sistema o aplicación que se encarga del procesamiento y gestión de datos.
- RestAPI: Arquitectura de programación que permite la comunicación entre diferentes sistemas mediante el uso del protocolo HTTP.
- Arquitectura de capas: Diseño de software que separa la lógica y las responsabilidades en diferentes capas o niveles.
- Protocolo HTTP: Protocolo de comunicación utilizado en la World Wide Web para el intercambio de información.
- Protocolo MQTT: Protocolo de mensajería ligero y basado en suscripción utilizado en aplicaciones de IoT.

Unidad 3 – Bases de datos

- Diseño de base de datos: Proceso de creación y estructuración de una base de datos para almacenar y gestionar información.
- Modelamiento de entidades de una solución: Creación de representaciones visuales que describen las entidades y relaciones de un sistema.
- Base de datos como componente de la red: Integración de una base de datos en una arquitectura de red para el almacenamiento y acceso a la información.

Unidad 4 – Consumo de servicios REST

- Páginas web como componente de la red: Uso de páginas web como interfaces de usuario en una arquitectura de red.
- Javascript: Lenguaje de programación utilizado para agregar interactividad a las páginas web.
- Comunicación con REST API desde Web: Intercambio de datos entre una aplicación web y una API REST mediante peticiones HTTP.
- Comunicación con REST API desde componentes de hardware: Interacción entre dispositivos de hardware y una API REST para el intercambio de datos.

Unidad 5 – Despliegue

- Despliegue local y remoto: Proceso de implementar y poner en funcionamiento una aplicación en un entorno local o en un servidor remoto.

- Uso de contenedores para distribuir la solución en la red local y remota: Utilización de tecnologías de contenedores como Docker para empaquetar y distribuir aplicaciones de manera eficiente.
- Docker: Plataforma de contenedores que permite empaquetar, distribuir y ejecutar aplicaciones en entornos aislados.

Metodologías de aprendizajes

El curso se desarrollará desde el punto de vista teórico en módulos de acuerdo con la descripción que aparece en el contenido. Sin embargo, se aplicará una metodología basada en proyectos donde se deben realizar las siguientes acciones:

- El profesor planteará un proyecto de desarrollo de software mediante un Backlog que va a contener las Historias Épicas de Usuario que describen el software a desarrollar. Se definirán roles, equipos y reglas del proyecto y del curso.
- El estudiante debe investigar sobre los temas propuestos por el profesor y elaborar presentaciones que demuestren su proceso de investigación con evidencias teóricas y prácticas. Estas actividades pueden involucrar la búsqueda de información con personas externas al aula y recursos digitales en diferentes fuentes de información. Todos los recursos usados deben ser referenciados en su trabajo.
- En clase se realizarán diferentes tipos de actividades que dependiendo del tema en que se esté avanzando se escogerán para ser aplicados o no:
 - a. Reunión inspirada en la Daily meeting de Scrum.
 - b. Reunión de planeación para el desarrollo de software.
 - c. Reunión de revisión y validación de los artefactos y del avance en el desarrollo del software.
 - d. Reunión de retrospectiva para evolucionar los equipos de trabajo y el proceso.

Se desarrollarán trabajos en grupo, en los cuales los estudiantes deberán aplicar todos los conocimientos adquiridos mediante los procesos de investigación propuestos y la guía del profesor.

Para lograr los objetivos de aprendizaje que se proponen, es necesario que cada estudiante realice las actividades que se presentan a continuación, en los momentos que se mencionan.

Evaluación de aprendizajes

Código evaluación	Mecanismo o actividad evaluativa	Porcentaje de la nota final	Relación con objetivos terminales - resultado de aprendizaje del curso	Relación con el resultado de aprendizaje de la competencia de egreso
PrAnálisis	PrAnálisis	10	OT1	SO1-PI1, SO1-PI2
PrBacklog	PrBacklog	10	OT3, OT4	SO1-PI3, SO5-PI1, SO5-PI2 , SO5-PI3
PrDiseño	PrDiseño	10	OT2	SO2-PI2, SO2-PI3
PrDoc	PrDoc	10	OT5	SO3-PI1, SO3-PI2

Código evaluación	Mecanismo o actividad evaluativa	Porcentaje de la nota final	Relación con objetivos terminales - resultado de aprendizaje del curso	Relación con el resultado de aprendizaje de la competencia de egreso
PrImplementen	PrImplementen	30	OT1, OT3	SO1-PI1, SO1-PI2, SO1-PI3, SO5-PI1, SO5-PI2, SO5-PI3
PrPropuest	PrPropuest	10	OT2	SO2-PI2, SO2-PI3
PrTEquipo	PrTEquipo	20	OT4	SO5-PI1, SO5-PI2, SO5-PI3

Recursos de apoyo

No es necesario comprar ningún libro para el curso. Todo el material necesario para el curso será suministrado a través de Intu, como los materiales bibliográficos, guías de clases, videos, blogs, cursos en línea y otros.

Algunos libros, enlaces y herramientas relevantes:

The Scrum Guide. (s.f.). Recuperado de <http://www.scrumguides.org/>

Pressman, R. (2010). Ingeniería del Software: Un Enfoque Práctico (7.a ed.). McGraw Hill.

Cloudflare. (s.f.). Internet Protocol. Recuperado de <https://www.cloudflare.com/es-es/learning/network-layer/internet-protocol/>

IBM. (s.f.). Internet Protocol - Transport Level. Recuperado de <https://www.ibm.com/docs/es/aix/7.1?topic=protocols-internet-transport-level>

IBM. (s.f.). Transmission Control Protocol. Recuperado de <https://www.ibm.com/docs/es/aix/7.1?topic=protocols-transmission-control-protocol>

Mavin, A. (s.f.). EARS. Recuperado de <https://alistairmavin.com/ears/>

IBM. (s.f.). User Datagram Protocol. Recuperado de <https://www.ibm.com/docs/es/aix/7.1?topic=protocols-user-datagram-protocol>

Cloudflare. (s.f.). User Datagram Protocol (UDP). Recuperado de <https://www.cloudflare.com/es-es/learning/ddos/glossary/user-datagram-protocol-udp/>

Atlassian. (s.f.). Introducción a la gestión de proyectos. Recuperado de <https://www.atlassian.com/es/agile/project-management/project-management-intro>

Atlassian. (s.f.). Flujo de trabajo en la gestión de proyectos. Recuperado de <https://www.atlassian.com/es/agile/project-management/workflow>

Atlassian. (s.f.). Épicos, historias y temas en la gestión de proyectos. Recuperado de <https://www.atlassian.com/es/agile/project-management/epics-stories-themes>