

**Código y nombre del curso:** 09749 – Ecosistemas de Aplicaciones

**Prerrequisito:** Diseñando con Algoritmos

**Correquisito:** Espacio de práctica de Ecosistemas de Aplicaciones

**Programa:** Diseño de Medios Interactivos

**Semestre:** Quinto semestre

**Periodo académico:** 2020-1

**Intensidad semanal:** 5 horas

**Créditos:** 3

**Es prerrequisito de:** Proyecto de Grado I - DMI

**Objetivo General**

Diseñar y desarrollar soluciones interactivas en un ecosistema digital y evaluar la usabilidad de acuerdo a requerimientos de un sistema.

# Objetivos Terminales

1. Identificar los requerimientos de un proyecto interactivo que requiera transmitir información entre nodos de un ecosistema digital.
2. Diseñar la arquitectura de una aplicación de un proyecto interactivo que permita la interacción en los nodos de un ecosistema digital.
3. Desarrollar aplicaciones dentro de un proyecto interactivo que permitan la interacción entre los nodos de un ecosistema digital.
4. Evaluar la usabilidad básica de un proyecto interactivo en un ecosistema digital.

# ESTRATEGIA PEDAGÓGICA

El curso tiene una metodología de trabajo iterativo cuyo eje principal es el proyecto final del curso. Durante el desarrollo de la clase se apropian conceptos que le permitirán trabajar de manera efectiva

Es responsabilidad del estudiante preparar el material general y el específico por disciplina, porque de eso depende el buen desarrollo de las sesiones con sus coequiperos. La asistencia es esencial.

**Específicos de formación académica**

**Unidad 1:**​ Introducción a la computación física

1.1. Identificar los requerimientos de la computación física basada en Arduino

1.2. Identificar las características de Arduino y los componentes del mismo diferenciando sus responsabilidades.

1.3. Utiliza herramientas de prototipado para planear los diseños físicos.

1.4. Crea aplicaciones que involucran la computación física a partir de Arduino.

**Unidad 2:** ​Principios básicos de redes

2.1. Identificar las topologías de red existentes, aplicaciones, ventajas y desventajas.  
2.2. Identificar las características del modelo OSI, los componentes del mismo y las responsabilidades de estos.

2.3. Describir la estructura, componentes y características del modelo IP.

2.4. Enumerar requerimientos funcionales y no funcionales.

2.5. Explicar el modelo cliente servidor.

2.6. Describir las características generales de Android como sistema operativo.  
2.7. Utilizar elementos propuestos por Android para la construcción de interfaces de usuario

**Unidad 3:** ​Aplicaciones en red utilizando el protocolo UDP

3.1. Identificar requerimientos de aplicaciones en red basadas en el protocolo UDP.  
3.2. Aplicar patrones de diseño en aplicaciones que usen UDP.

3.3. Desarrollar aplicaciones que hagan uso del protocolo UDP.

3.4. Evaluar usabilidad en aplicaciones basadas en el protocolo de comunicación UDP.  
3.5. Establecer mecanismos de comunicación asíncrona.

3.6. Utilizar elementos avanzados propuestos por Android para la construcción de interfaces de usuario.

3.7. Crear aplicaciones que aprovechen multicast como concepto de comunicación.

**Unidad 4:** ​Aplicaciones en red utilizando el protocolo TCP

4.1. Identificar requerimientos de aplicaciones en red basadas en el protocolo TCP.

4.2. Aplicar patrones de diseño en aplicaciones que usen TCP.

4.3. Desarrollar aplicaciones que hagan uso del protocolo TCP.

4.4. Evaluar usabilidad en aplicaciones basadas en el protocolo de comunicación TCP.

4.5. Establecer mecanismos de comunicación síncrona.

4.6. Crear aplicaciones de servidor que funcionan con múltiples hilos.

4.7. Generar persistencia a través del uso del estándar JSON.

4.8. Aplicar serialización para el envío de objetos compuestos.

4.9. Diseñar bases de datos NO-SQL.

**Unidad 5:** ​Aplicaciones en red utilizando Firebase y MQTT - Internet de las Cosas

5.1.  Identificar requerimientos de aplicaciones en red basadas el concepto de Internet de las cosas.

5.2.  Aplicar patrones de diseño en aplicaciones que usen Firebase/MQTT para posibilitar el concepto Internet de las cosas.

5.3.  Desarrollar aplicaciones que posibiliten la conexión de los nodos de un ecosistema haciendo uso de Firebase/MQTT.

5.4.  Evaluar usabilidad en aplicaciones basadas en el concepto Internet de las cosas que usan Firebase/MQTT.

5.5.  Crear Aplicaciones para Arduino que incorporen componentes de conexión a la red.

# EVALUACIÓN

|  |  |
| --- | --- |
| ***Concepto*** | ***Porcentaje*** |
| Parcial 1 | 15% |
| Parcial 2 | 15% |
| Taller individual 1 | 15% |
| Taller individual 2 | 15% |
| Quices teóricos | 10% |
| Quices prácticos | 5% |
| Ejercicios en clase y tareas | 5% |
| Proyecto final | 20% |
| **TOTAL** | 100 % |

Durante el semestre se calificará solo 3 talleres de ejercicios en clase que corresponderán al porcentaje de las Pruebas cortas y tareas junto con los quices. Se recomienda ser muy puntuales con las entregas de estos talleres y su correcto desarrollo, al final del semestre se calificará una consolidado de todos los ejercicios propuestos.

●  La no entrega de ficha técnica acarrea una nota de cero (0.0) en el componente de diseño del taller completo.

●  La no entrega de diagrama de clases UML acarrea una nota de cero (0.0) en el componente de programación del taller completo.

●  Solo se tendrá́ en cuenta para evaluación lo que esté en Moodle.

●  Los talleres y sus componentes (Fichas técnicas, UML y Código) tendrá́ dos fechas de entrega: entrega ​**esperada** (que es cuando debería entregarse) y entrega ​**máxima** (que es la fecha máxima que se recibe el trabajo para ser calificado). Lo que no se entregue antes de la entrega máxima, NO se califica.

●  Las fechas ​**esperada** y ​**máxima** tienen una diferencia de 4 días entre ellas.

Eso significa que si alguien entrega el trabajo en el día 3, tendrá́ una nota máxima de 4,5 y el que entregue el día de la fecha máxima tendrá́ una nota sobre 4,0.

●  Recordar que lo que no está en Moodle no existe, por lo tanto, no se recibirán entregas por correo ni otros medios.

**PROFESOR:**

Domiciano Rincón: domiciano.rincon@correounivalle.edu.co

**Bibliografía**

* Java Network Programming - O'Really - Cap 2, 3, 8, 12
* Conceptos Avanzados de programación en Red - Universidad Icesi
* The encyclopedia of Human Computer Interaction 2nd Edition - Interaction Design Foundation – Cap 13
* Head First Design Patterns
* Margolis, M. (2012). Arduino cookbook: 2. ed. O'Reilly Media, Inc.
* Monk, S. (2012). Arduino + Android projects for the evil genius. McGraw Hill.
* Norris, D. (2013). The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and BeagleBone Black. DND Ventures LLC.
* Rose, D. (2014). Enchanted Objects: Design, Human Desire, and the Internet of Things. Scribner.
* Kellmereit, D., & Obodovski, D. (2013). The Silent Intelligence: The Internet of Things. DND Ventures LLC.

**Webgrafía**

https://developer.android.com/develop/index.html

https://docs.oracle.com/javase/tutorial/networking/index.html

https://docs.oracle.com/javase/8/docs/technotes/guides/net/

https://developer.android.com/studio/intro/index.html?hl=es-419

https://developer.android.com/reference/android/app/ListActivity.html

https://developer.android.com/reference/android/content/Intent.html

https://docs.oracle.com/javase/tutorial/jndi/objects/serial.html

http://docs.oracle.com/javase/tutorial/essential/io/buffers.html

https://firebase.google.com/docs/android/setup

https://docs.oracle.com/javase/tutorial/jaxp/