



Universidad Nacional Autónoma  
de México  
Facultad de Ciencias



Computación Distribuida 2026-1

## Práctica 1

Baños Mancilla Ilse Andrea  
321173988

Rivera Machuca Gabriel Eduardo  
321057608

# Reporte

Candidato: RUP

Yo elegiría el modelo RUP debido a que es bueno para equipos grandes, además que menciona que es un equipo grande de 15 personas lo cual ayuda a amortiguar lo pesado en conjunto a que es un trabajo con un plazo de 18 meses, entonces por eso, para mi RUP es el modelo ideal.

En esta práctica desarrollamos algoritmos y estructuras básicas en la computación distribuida. Para empezar a trabajar con la comunicación entre nodos, implementamos interfaces como Nodo y Canal Broadcast, esto es necesario para que los nodos intercambien información en forma de mensajes a través de los canales. Por eso mismo, se implementa CanalComunicación.

## Primer algoritmo

El primer algoritmo fue NodoVecinos. Esto se logró gracias al pseudocódigo proporcionado en el documento con las instrucciones. Esta clase incluye el método conoceVecino que se encarga de enviar la lista de los vecinos del nodo usando el canal de comunicación, ese nodo se queda esperando mensajes en el canal de entrada, si se recibe algún mensaje actualiza el conjunto de identificadores.

## Segundo algoritmo

El segundo algoritmo fue NodoTopologia. Para esta clase se modificó el constructor de la clase Nodo, en este recibe el identificador del nodo, lista de vecinos, canal de entrada, canal de salida y mensaje. El algoritmo inicializa el conjunto de procesos conocidos con el identificador del nodo y se construye el conjunto de canales conocidos con las conexiones entre el nodo y sus vecinos. De igual manera, el algoritmo se logró implementar gracias al pseudocódigo proporcionado en el documento. Ya que el nodo conoce a sus vecinos, registra a las aristas y envía mensaje a los vecinos, después envía mensaje a todos los vecinos, si no tenía información se actualiza y guarda, después reenvía el mensaje a todos menos al emisor del mensaje.

## Tercer Algoritmo

Igual se modificó la clase Nodo, agregando seen\_message, este booleano lo usamos para decir si el nodo ya ha recibido un mensaje. Igual el algoritmo se logró con el tercer pseudocódigo del documento. Aquí el nodo\_0 recibe un mensaje y lo marcamos como visto en seen\_message y lo envía al nodo vecino usando el nodo\_0 como el nodo fuente para propagar el mensaje, así se repite, el vecino que recibe el mensaje ahora es la fuente y lo propaga a sus vecinos, si un vecino recibe el mensaje repetido solo lo ignora.