

Proyecto Final - Sistema de Noticias

Sistemas Operativos



Daniel Felipe Castro Moreno

Eliana Katherine Cepeda Gonzalez

María Paula Rodríguez Ruiz

Daniel Horacio Gonzalez Orduz

Profesor:

John Corredor Franco

Dpto. de Ingeniería de Sistemas

Pontificia Universidad Javeriana

Bogotá D.C.

11 de Noviembre del 2024

Contenido

Introducción	3
Arquitectura	3
Estructuras De Datos.....	4
Mecanismos De Comunicación	5
Flujo De Mensajes.....	5
Sincronización Y Concurrencia.....	6
Terminación Del Programa	6
Diagrama De Secuencia.....	7
Plan De Pruebas	11

Introducción

El objetivo de este proyecto es construir un *Sistema de Noticias* bajo el patrón Publicador/Suscriptor, por lo que para el diseño se propondrá lo siguiente con el fin de conseguir una comunicación eficiente entre procesos, usando mecanismos de sincronización apropiados y permitiendo la extensibilidad del sistema para añadir más publicadores y suscriptores sin necesidad de modificar significativamente la estructura del SC.

Arquitectura

Para la arquitectura general se propone lo siguiente:

- **Publicadores (P):** Son procesos que publican noticias a través de un *pipe nominal* (pipePSC) hacia el *Sistema de Comunicación (SC)*. Cada publicador lee noticias de un archivo con un formato específico y las envía en intervalos definidos por el tiempo timeN.
- **Sistema de Comunicación (SC):** Actúa como intermediario entre publicadores y suscriptores, recibiendo noticias y entregándolas a los suscriptores correctos. Es el núcleo del sistema, gestionando tanto la recepción de noticias como las suscripciones.
- **Suscriptores (S):** Procesos que se suscriben a uno o más tipos de noticias a través de pipeSSC y de un pipe único temporal propio. Esperan recibir noticias que coincidan con los temas de su interés.

Estructuras de datos

- **Publicador:**
 - Almacena temporalmente cada noticia leída del archivo en el buffer
buffer[BUFFER_SIZE]. Este es un arreglo de caracteres.
- **Suscriptor:**
 - Un arreglo Suscriptor suscriptores[MAX_SUSCRIPTORES] permite almacenar hasta 100 suscriptores registrados.
 - La variable num_suscriptores lleva el conteo de los suscriptores registrados.
- **Sistema de Comunicación:**
 - La estructura Suscriptor representa a cada suscriptor con sus temas de interés y el nombre de su tubería. Incluye:
 - topics: que contiene una lista de temas en los que está interesado el suscriptor.
 - pipe_name: el nombre de la tubería exclusiva para enviar noticias al suscriptor.
 - Usa la estructura Noticia que contiene:
 - tipo: el tipo o categoría de la noticia (el primer carácter del mensaje).
 - contenido: el texto completo de la noticia.
 - Un arreglo Noticia noticias[MAX_NOTICIAS] guarda hasta 100 noticias recibidas y almacenadas.
 - La variable num_noticias cuenta el número de noticias almacenadas.

Mecanismos de comunicación

Para los mecanismos de comunicación se proponen los *pipes* nominales (pipePSC y pipeSSC) para el flujo de datos entre publicadores y el SC, así como entre el SC y los suscriptores además de nuevos pipes temporales creados específicamente para cada nuevo suscriptor.

Flujo de mensajes

Ahora bien, en cuanto al flujo de mensajes se propone lo siguiente para nuestra arquitectura general:

- **Publicador:**
 1. Lee el archivo de noticias.
 2. Valida el formato de cada línea.
 3. Envía cada noticia a través de pipePSC al SC.
- **Sistema de Comunicación (SC):**
 1. Recibe noticias de los publicadores.
 2. Verifica la categoría de cada noticia y consulta la lista de suscriptores interesados.
 3. Envía la noticia a través de los pipes únicos temporales a los suscriptores correspondientes registrados en el pipe SSC.
 4. Monitorea el tiempo timeF y notifica el fin de las emisiones.
- **Suscriptor:**
 1. Se suscribe a temas de interés a través de pipeSSC.
 2. Recibe noticias relevantes a través de su pipe único.

3. Muestra las noticias en su pantalla hasta que el SC notifique el fin de las transmisiones.

Sincronización y concurrencia

La sincronización y concurrencia en el proyecto se maneja a través de la creación dinámica de publicadores y suscriptores. Cada proceso se administra para aceptar y manejar múltiples conexiones a través del SC que actúa como coordinador general. Para gestionar la finalización de estos procesos y coordinar la recepción de nuevas conexiones, se utiliza el mecanismo de señales (signal). Específicamente:

- **Señales de Finalización:** Se envían señales específicas para terminar publicadores y suscriptores de manera controlada, evitando conflictos o problemas de recursos bloqueados.
- **Manejo de Conexiones:** El SC está configurado para recibir y asignar nuevos suscriptores y publicadores en forma dinámica, permitiendo la escalabilidad y adaptación en tiempo real a las solicitudes entrantes.

Terminación del Programa

El programa finaliza en los siguientes pasos:

1. **Desconexión de Clientes Activos:** Primero, el SC envía una señal a todos los procesos publicadores y suscriptores activos, indicándoles que deben concluir sus actividades y cerrar cualquier conexión abierta.

2. **Liberación de Recursos:** Una vez terminados los procesos, el SC asegura la liberación de memoria dinámica y otros recursos utilizados, como sockets o descriptores de archivo, para evitar fugas de memoria.
3. **Confirmación de Terminación:** El SC verifica que todos los procesos hayan finalizado correctamente antes de dar por concluida su propia ejecución, cerrando cualquier hilo o conexión restante.

Diagrama de secuencia

El diagrama de secuencia del sistema funciona con tres tipos pipes (canales de comunicación interproceso): pipePSC para mensajes del publicador al sistema, pipeSSC para mensajes del suscriptor al sistema, y un pipeUnique temporal para la transmisión de noticias a cada suscriptor. Cabe resaltar que existen tantos pipes temporales como suscriptores y que en cualquier momento se puede realizar una nueva suscripción o publicación.

Fases principales del diagrama

1. **Inicialización:**
 - Se define una estructura para almacenar múltiples suscriptores y otra para almacenar las noticias en el sistema de comunicación.
 - Se crean los pipes pipePSC y pipeSSC con permisos (0666) para comunicación entre el sistema y los actores externos.
 - El sistema abre pipePSC y pipeSSC en modo de solo lectura y sin bloqueo.

- El publicador abre pipePSC en modo de solo escritura y el suscriptor abre pipeSSC en modo de solo lectura.
- Se crea un pipeUnique adicional, exclusivo para cada transmisión de noticias a los suscriptores, y el suscriptor lo abre en modo de solo lectura.
- Se establece un timeout (con un valor timeF) que sirve como límite para esperar eventos en el sistema de comunicación.

2. Ciclo de Recepción y Transmisión:

- Este ciclo se ejecuta continuamente mientras el sistema esté activo.
- El sistema reinicia los file descriptors (FD) y agrega a la lista de monitoreo los pipes pipePSC y pipeSSC.
- **Recepción de Mensajes:**
 - Si hay un mensaje en pipePSC (indicador de que el publicador envió una noticia):
 - El publicador escribe la noticia en pipePSC, el sistema la lee, y luego la almacena.
 - Si hay un mensaje en pipeSSC (indicador de que el suscriptor envió una solicitud):
 - El suscriptor escribe la solicitud en pipeSSC, el sistema la lee, y luego registra al suscriptor.
- **Transmisión de Noticias:**
 - Para cada noticia almacenada, el sistema verifica si coincide con los temas de interés del suscriptor.

- Si coincide, el sistema envía la noticia a través de pipeUnique al suscriptor.
- El suscriptor lee el mensaje de pipeUnique y lo muestra por pantalla.

3. Finalización de Transmisión:

- Si el tiempo de espera se agota, el sistema finaliza la transmisión.
- Para cada suscriptor, el sistema envía un mensaje de "Fin de transmisión" a través de pipeUnique.
- El suscriptor lee este mensaje, lo muestra, y luego cierra y elimina pipeUnique.

4. Cierre del Sistema:

- Después de salir del ciclo principal, el sistema cierra y elimina los pipes pipePSC y pipeSSC, terminando la comunicación con el publicador y el suscriptor.

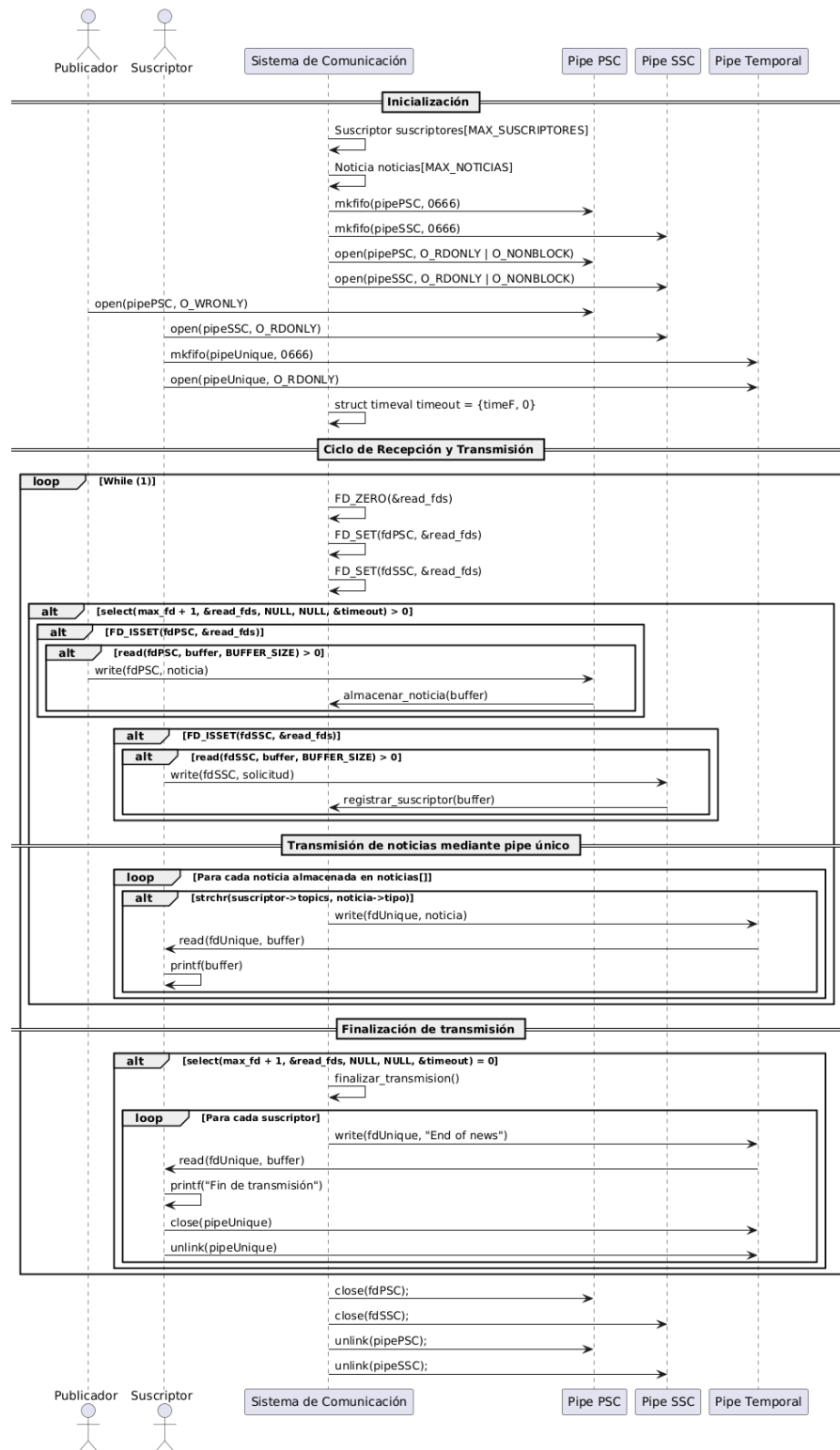


Figura 1. Diagrama de secuencia del sistema. Elaboración propia.

Plan de pruebas

Planificación de pruebas:

Previo a la codificación fue realizado un proceso de análisis respecto al funcionamiento del sistema de comunicación para de esa forma establecer que pruebas podrían ser claves para evaluar el correcto funcionamiento y calidad del código elaborado, por lo que se planteo lo siguiente:

1. Realizar una prueba en la que existieran dos publicadores y un suscriptor donde se evaluara la correcta recepción y envío de diferentes noticias junto a la correcta finalización del sistema.
2. Realizar una prueba que validara si efectivamente se rechazaban las noticias que tenían un formato incorrecto, desde utilizar un tópico diferente a (A, E, C, P, S), tener mas de 80 caracteres, no terminar con punto o incluso no llevar los dos puntos seguidos del tópico.
3. Realizar una prueba en la que existieran dos suscriptores y dos publicadores para evaluar que efectivamente a cada uno de ellos se les enviaran las noticias correspondientes sin falla alguna en la sincronización o manejo del sistema.

Ejecución de pruebas:

Siguiendo la estrategia presentada en la planeación de pruebas, se ejecutaron de acuerdo a su especificación y los resultados fueron los siguientes:

```
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$ cd SO/Proyecto_Sistema_Noticias/
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$ ls
Makefile README.md data src
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$ make
mkdir -p build
gcc -Wall -g -c src/publicador.c -o build/publicador.o
gcc -Wall -g -c src/utills.c -o build/utills.o
gcc -Wall -g -o publicador build/publicador.o build/utills.o
gcc -Wall -g -c src/suscriptor.c -o build/suscriptor.o
gcc -Wall -g -o suscriptor build/suscriptor.o build/utills.o
gcc -Wall -g -c src/sistema_comunicacion.c -o build/sistema_comunicacion.o
gcc -Wall -g -c src/SCBiblioteca.c -o build/SCBiblioteca.o
gcc -Wall -g -o sistema build/sistema_comunicacion.o build/SCBiblioteca.o build/utills.o
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$ ls
Makefile README.md build data publicador sistema src suscriptor
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$ ./sistema -p pipePSC -s pipeSSC -t 10
>>> SYSTEM STARTED <<<
News received from publisher: A: Exposición de arte moderno en el museo local.
News received from publisher: P: Las elecciones se celebrarán el próximo mes.
Subscriber registered: Pipe=/tmp/pipeS_1100 Topics=A

News received from publisher: E: La famosa banda de rock se presenta este fin de semana
.
News received from publisher: C: Científicos descubren una nueva especie de insecto.
News received from publisher: S: Robo en una joyería del centro comercial.
News received from publisher: C: Nueva misión espacial a Marte programada para 2025.
News received from publisher: A: Artista local lanza su nueva galería virtual.
News received from publisher: P: El congreso debate nuevas reformas.
News received from publisher: E: Entrevista exclusiva con el director de la nueva película.
News received from publisher: S: Incendio en el parque natural.
No new publishers detected. Broadcasting end of transmission.
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$
```

```
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$ ./publicador -p pipePSC -f data/noticias1.txt -t 3
Noticia enviada: A: Exposición de arte moderno en el museo local.
Noticia enviada: P: Las elecciones se celebrarán el próximo mes.
Noticia enviada: E: La famosa banda de rock se presenta este fin de semana.
Noticia enviada: C: Científicos descubren una nueva especie de insecto.
Noticia enviada: S: Robo en una joyería del centro comercial.
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$ ./publicador -p pipePSC -f data/noticias2.txt -t 3
Noticia enviada: C: Nueva misión espacial a Marte programada para 2025.
Noticia enviada: A: Artista local lanza su nueva galería virtual.
Noticia enviada: P: El congreso debate nuevas reformas.
Noticia enviada: E: Entrevista exclusiva con el director de la nueva película.
Noticia enviada: S: Incendio en el parque natural.
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$
```

```
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$ ./suscriptor -s pipeSSC
Enter topics of interest (e.g., A E S): A
Waiting for news on selected topics...
Received news: A: Exposición de arte moderno en el museo local.
Received news: A: Artista local lanza su nueva galería virtual.
End of news broadcast.
crackmaps@Crackmaps:~/SO/Proyecto_Sistema_Noticias$
```

Figura 2. Captura prueba 1 – Dos publicadores, un suscriptor. Elaboración propia.

```
GNU nano 6.2 data/noticias1.txt *
A: Exposición de arte moderno en el museo local.
b: Las elecciones se celebrarán el próximo mes.
E: La famosa banda de rock se presenta este fin de semana
C: Científicos descubren una nueva especie de insecto.
S: Robo en una joyería del centro comercial.
```

Figura 3. Captura prueba 2 – Modificación al archivo de noticias para mostrar los 3 casos en que debe rechazar su envío. Elaboración propia.

```

crackmaps@Crackmaps:~/S0/Proyecto_Sistema_Noticias$ nano data/noticias1.txt
crackmaps@Crackmaps:~/S0/Proyecto_Sistema_Noticias$ ./sistema -p pipePSC -s pipeSSC -t
10
>>> SYSTEM STARTED <<<
News received from publisher: C: Nueva misión espacial a Marte programada para 2025.
News received from publisher: A: Artista local lanza su nueva galería virtual.
Subscriber registered: Pipe=/tmp/pipeS_1852 Topics=A

News received from publisher: P: El congreso debate nuevas reformas.
News received from publisher: E: Entrevista exclusiva con el director de la nueva pelí-
cula.
News received from publisher: S: Incendio en el parque natural.
News received from publisher: A: Exposición de arte moderno en el museo local.
News received from publisher: C: Científicos descubren una nueva especie de insecto.
No new publishers detected. Broadcasting end of transmission.
crackmaps@Crackmaps:~/S0/Proyecto_Sistema_Noticias$

```

Figura 4. Captura prueba 2 – Salida del sistema. Elaboración propia.

```

crackmaps@Crackmaps:~/S0/Proyecto_Sistema_Noticias$ ./publicador -p pipePSC -f da
ta/noticias2.txt -t 3
Noticia enviada: C: Nueva misión espacial a Marte programada para 2025.
Noticia enviada: A: Artista local lanza su nueva galería virtual.
Noticia enviada: P: El congreso debate nuevas reformas.
Noticia enviada: E: Entrevista exclusiva con el director de la nueva película.
Noticia enviada: S: Incendio en el parque natural.
crackmaps@Crackmaps:~/S0/Proyecto_Sistema_Noticias$ ./publicador -p pipePSC -f da
ta/noticias1.txt -t 3
Noticia enviada: A: Exposición de arte moderno en el museo local.
Invalid news format (missing period or other error): b: Las elecciones se celebra
rán el próximo mes.
Invalid news format (missing period or other error): E: La famosa banda de rock s
e presenta este fin de semana
Noticia enviada: C: Científicos descubren una nueva especie de insecto.
Invalid news format (missing period or other error): S Robo en una joyería del ce
ntro comercial.
crackmaps@Crackmaps:~/S0/Proyecto_Sistema_Noticias$

```

Figura 5. Captura prueba 2 – Salida del publicador. Elaboración propia.

```

crackmaps@Crackmaps:~/S0/Proyecto_Sistema_Noticias$ ./suscriptor -s pipeSSC
Enter topics of interest (e.g., A E S): A
Waiting for news on selected topics...
Received news: A: Artista local lanza su nueva galería virtual.
Received news: A: Exposición de arte moderno en el museo local.
End of news broadcast.
crackmaps@Crackmaps:~/S0/Proyecto_Sistema_Noticias$

```

Figura 6. Captura prueba 2 – Salida del suscriptor. Elaboración propia.

```

crackmaps@Crackmaps: ~/SC
crackmaps@Crackmaps:~/SC$ ./sistema -p pipePSC -s pipeSSC -t 10
>>> SYSTEM STARTED <<<
News received from publisher: A: Exposición de arte moderno en el museo local.
News received from publisher: P: Las elecciones se celebrarán el próximo mes.
Subscriber registered: Pipe=/tmp/pipeS_3028 Topics=A

News received from publisher: E: La famosa banda de rock se presenta este fin de semana
.
News received from publisher: C: Científicos descubren una nueva especie de insecto.
Subscriber registered: Pipe=/tmp/pipeS_3046 Topics=C

News received from publisher: S: Robo en una joyería del centro comercial.
News received from publisher: A: Artista local lanza su nueva galería virtual.
News received from publisher: P: El congreso debate nuevas reformas.
News received from publisher: E: Entrevista exclusiva con el director de la nueva película.
News received from publisher: S: Incendio en el parque natural.
No new publishers detected. Broadcasting end of transmission.
crackmaps@Crackmaps:~/SC/Proyecto_Sistema_Noticias$

crackmaps@Crackmaps:~/SC$ ./publicador -p pipePSC -f data/noticias1.txt -t 3
Noticia enviada: A: Exposición de arte moderno en el museo local.
Noticia enviada: P: Las elecciones se celebrarán el próximo mes.
Noticia enviada: E: La famosa banda de rock se presenta este fin de semana.
Noticia enviada: C: Científicos descubren una nueva especie de insecto.
Noticia enviada: S: Robo en una joyería del centro comercial.
crackmaps@Crackmaps:~/SC/Proyecto_Sistema_Noticias$ ./publicador -p pipePSC -f data/noticias2.txt -t 3
Noticia enviada: C: Nueva misión espacial a Marte programada para 2025.
Noticia enviada: A: Artista local lanza su nueva galería virtual.
Noticia enviada: P: El congreso debate nuevas reformas.
Noticia enviada: E: Entrevista exclusiva con el director de la nueva película.
Noticia enviada: S: Incendio en el parque natural.
crackmaps@Crackmaps:~/SC/Proyecto_Sistema_Noticias$

crackmaps@Crackmaps:~/SC/Proyecto_Sistema_Noticias$ ./suscriptor -s pipeSSC
crackmaps@Crackmaps:~/SC/Proyecto_Sistema_Noticias$ ./suscriptor -s pipeSSC
Enter topics of interest (e.g., A E S): A
Waiting for news on selected topics...
Received news: A: Exposición de arte moderno en el museo local.
Received news: A: Artista local lanza su nueva galería virtual.
End of news broadcast.
crackmaps@Crackmaps:~/SC/Proyecto_Sistema_Noticias$

crackmaps@Crackmaps:~/SC/Proyecto_Sistema_Noticias$ ./suscriptor -s pipeSSC
crackmaps@Crackmaps:~/SC/Proyecto_Sistema_Noticias$ ./suscriptor -s pipeSSC
Enter topics of interest (e.g., A E S): C
Waiting for news on selected topics...
Received news: C: Científicos descubren una nueva especie de insecto.
Received news: C: Nueva misión espacial a Marte programada para 2025.
End of news broadcast.
crackmaps@Crackmaps:~/SC/Proyecto_Sistema_Noticias$

```

Figura 7. Captura prueba 3 – Dos suscriptores, dos publicadores. Elaboración propia.

Es así como concluye nuestro plan de pruebas donde se puede evidenciar visualmente a lo largo de las 3 pruebas como no solo el programa es completamente funcional sino que también es estético y con calidad, por lo que se concluye exitosamente el desarrollo del proyecto.