

Para el proyecto de paralelismo del curso vamos a construir un cluster de computación en Elixir, compuesto por Raspberry Pi. Para ello, utilizaremos nerves (<https://nerves-project.org>) un proyecto para utilizar Elixir en sistemas embebidos.

Como primera medida, es necesario montar Elixir sobre cada Raspberry Pi. Para ello toca instalar Nerves dentro de la memoria SD que booteará el Raspberry Pi. Las instrucciones de instalación de Nerves para todos los sistemas operativos se encuentran <https://hexdocs.pm/nerves/installation.html>, en MacOS es

1. Descargue e instale el manejador de paquetes fwup - <https://github.com/fwup-home/fwup> - <https://github.com/fwup-home/fwup#installing>. `brew install fwup` funciona suficientemente bien en MAC.
2. Descargue el firmware apropiado para el Raspberry Pi (`nerves_livebook_rpi4.fw`), adjunto a los documentos del proyecto
3. Ejecute el firmware dentro de la tarjeta SD del Raspberry Pi. Para hacer esto, ubíquese en la carpeta en la que se encuentra el firmware y ejecute el comando `sudo NERVES_WIFI_SSID='access_point' NERVES_WIFI_PASSPHRASE='passphrase' fwup nerves_livebook_rpi4.fw`. Las variables opcionales del WIFI se utilizan para poder darle capacidades de internet al Raspberry Pi y poder comunicarse con él por ahí, si es necesario.

Ya en el Raspberry Pi, es posible correr código de Elixir directamente desde la consola. Para crear nuevos proyectos, se pueden crear proyectos de nerves con mix y poder ejecutarlos directamente se utiliza livebook <https://hexdocs.pm/nerves/getting-started.html#nerves-livebook>.

Task 1. El proyecto consiste en la creación como tal del cluster, sobre el cual se van a ejecutar diferentes tareas en paralelo. Para ello deben:

- Se debe crear un cluster de Raspberry Pis conectando (por medio de cables ethernet). De tal forma que se puedan ejecutar tareas dentro del cluster en paralelo.
- Crear un repositorio con el código de la interfaz del cluster, que permita crear y manejar el cluster con tantos (o tan pocos) Raspberry Pi como queramos. Adicionalmente la interfaz debe ser capaz de recibir las tareas a ejecutar y correrlas dentro del cluster.
- La calificación del proyecto tendrá en cuenta el funcionamiento del cluster y al colaboración de cada uno de los grupos dentro del mismo, por medio de commits/merges/pull requests.

-
- Finalmente, el proyecto tendrá una sustentación, donde ejecutaremos tareas en paralelo en el cluster, los resultados de las tareas para cada grupo deben representar una mejora en el tiempo de ejecución. Dentro de la sustentación, cada grupo deberá ser capaz de calcular el speed up y el throughput de su solución con respecto a los datos obtenidos dentro de las tareas de programación correspondientes (tarea de programación en paralelo).

El proyecto se debe realizar en grupos (de dos o tres integrantes) de acuerdo a la asignación de los dispositivos. Adicionalmente existe el componente de trabajo entre todo el salón en el que todos los grupos deben participar.