Este proyecto tiene:

* RAN
* IA
* …….

RAN significa "Red de Acceso por Radio" (Radio Access Network en inglés). Es la parte de la red de telecomunicaciones móviles que se encarga de establecer la conexión inalámbrica entre los dispositivos móviles y la red principal. La RAN incluye las estaciones base, antenas, transmisores y receptores que forman la red inalámbrica y que proporcionan acceso a los servicios de comunicación móvil, como llamadas de voz, mensajes de texto y acceso a internet.

La RAN es esencial para la comunicación móvil y es una de las partes más visibles de la infraestructura de red móvil. Permite que los dispositivos móviles se conecten a la red de forma inalámbrica y proporciona la cobertura de la red móvil. Los diferentes tipos de tecnologías de comunicación móvil, como GSM, UMTS, LTE y 5G, utilizan diferentes tipos de RAN.

En general, la RAN es responsable de la comunicación inalámbrica entre los dispositivos móviles y la red de telecomunicaciones, mientras que la parte central de la red se encarga de la conexión entre diferentes partes de la red y de la conexión a internet y otros servicios de red.

En este proyecto, la RAN será una red 4G (LTE), montar este tipo de redes es un proyecto complejo que involucra varios componentes técnicos y financieros. Aquí hay algunos de los elementos principales que se necesitan para montar una red 4G:

1. Licencias: Antes de construir una red 4G, se requiere obtener una licencia de espectro de frecuencia de un organismo regulador. La adquisición de una licencia puede ser un proceso costoso y competitivo, y la disponibilidad de licencias puede variar según la ubicación geográfica.
2. Infraestructura de red: Para implementar una red 4G, se necesita una infraestructura de red, que incluye torres de antena, equipos de transmisión y conmutación, y sistemas de respaldo de energía. Esta infraestructura puede ser costosa y requiere un diseño y planificación cuidadosos.
3. Equipo de red: Además de la infraestructura de red, se necesitan equipos de red, como los equipos de radio de acceso (RAN) y los equipos de núcleo de red (core network) para conectarse con otros proveedores de red y permitir el acceso a los usuarios.
4. Personal técnico: Montar una red 4G requiere un equipo técnico capacitado en planificación, diseño, implementación y mantenimiento de redes de telecomunicaciones.
5. Costos financieros: Implementar una red 4G es un proyecto costoso y requiere una inversión significativa en capital, desde la adquisición de licencias hasta la infraestructura de red y los costos de personal técnico.
6. Certificación y cumplimiento de estándares: Para operar una red 4G, se requiere cumplir con los estándares de la industria y obtener la certificación necesaria.

En resumen, montar una red 4G es un proyecto complejo y costoso que involucra múltiples componentes técnicos y financieros. Se requiere una planificación cuidadosa y una inversión significativa para asegurar que la red sea efectiva y cumpla con los estándares de la industria.

**Vamos con la parte de software**

Existen varios software para implementar una RAN (Red de Acceso por Radio) en diferentes tecnologías de comunicación móvil. Aquí se presentan algunas de las principales:

1. OpenAirInterface (OAI): Es un proyecto de software libre y de código abierto que proporciona una implementación completa de una RAN para 4G LTE y 5G NR (New Radio). OAI es un software altamente flexible y configurable, y permite una rápida implementación y prototipado de nuevas ideas y tecnologías. Está diseñado para funcionar en hardware de propósito general y es compatible con hardware comercial.
2. srsRAN: Es otro proyecto de software libre y de código abierto que proporciona una implementación de RAN para tecnologías de comunicación móvil, como GSM, UMTS, LTE y 5G. srsRAN utiliza tecnologías de radio definidas por software (SDR) y es altamente modular y escalable. srsRAN se basa en hardware de propósito general y es una alternativa a los equipos de radio propietarios costosos y cerrados.
3. NextEPC: Es una solución de software de código abierto para la red de núcleo de paquetes (EPC) de LTE y 5G. NextEPC se integra con OpenAirInterface y otros proyectos de RAN de código abierto para proporcionar una solución completa de RAN y núcleo para 4G y 5G.
4. Ericsson Radio System: Es un software comercial de RAN para tecnologías de comunicación móvil como 2G, 3G, 4G y 5G. Es una solución completa de RAN que incluye hardware y software propietarios. Ericsson Radio System es ampliamente utilizado en todo el mundo por operadores de telecomunicaciones y proveedores de servicios.
5. Huawei Radio Access Network: Es otra solución comercial de RAN para tecnologías de comunicación móvil como 2G, 3G, 4G y 5G. Es una solución completa de RAN que incluye hardware y software propietarios. Huawei Radio Access Network también es ampliamente utilizado en todo el mundo por operadores de telecomunicaciones y proveedores de servicios.

Los dos softwares recomendados en clase fueron los dos primeros, OpenAirInterface (OAI) y srsRAN, ambas son dos soluciones diferentes para implementar redes de acceso de radio, que tienen sus propias ventajas y desventajas, y la elección depende de las necesidades específicas de cada organización.

Aquí hay algunas ventajas para elegir srsRAN:

1. Software libre y abierto: srsRAN es un proyecto de software libre y abierto que permite a los usuarios modificar y adaptar el software para satisfacer sus necesidades. Esto permite una mayor flexibilidad y control sobre la red de acceso de radio.
2. Soporte de múltiples interfaces: srsRAN admite una amplia gama de interfaces, lo que lo hace más versátil en términos de interoperabilidad con otras redes y tecnologías.
3. Escalabilidad: srsRAN puede escalar fácilmente para soportar grandes redes de acceso de radio y su arquitectura modular permite la adición de nuevas funcionalidades y características.

Por otro lado, OpenInterface también tiene sus propias ventajas:

1. Simplifica la implementación de redes: OpenInterface se centra en la simplificación de la implementación de redes de acceso de radio y en la eliminación de complejidades innecesarias.
2. Enfoque en las necesidades del cliente: OpenInterface está diseñado para satisfacer las necesidades específicas del cliente y se puede adaptar para trabajar con una amplia gama de dispositivos y tecnologías.
3. Conformidad con los estándares: OpenInterface está diseñado para cumplir con los estándares de la industria y para asegurar la interoperabilidad con otras redes y tecnologías.

Sin embargo, también presentan diferentes desventajas:

Desventajas de OpenInterface:

1. Limitaciones en la personalización: a diferencia de srsRAN, que es un software libre y abierto, OpenInterface puede tener limitaciones en cuanto a la personalización del software para adaptarse a las necesidades específicas de una organización.
2. Menos soporte de interfaces: aunque OpenInterface admite una amplia gama de dispositivos y tecnologías, puede tener menos soporte de interfaces en comparación con srsRAN.
3. Mayor complejidad en la configuración: OpenInterface se enfoca en la simplificación de la implementación de redes, pero la configuración puede ser más compleja en comparación con srsRAN, especialmente para usuarios no técnicos.

Desventajas de srsRAN:

1. Menos soporte de dispositivos: aunque srsRAN admite una amplia gama de interfaces, puede tener menos soporte de dispositivos en comparación con OpenInterface.
2. Menos documentación: en comparación con OpenInterface, que tiene una amplia documentación en línea, puede haber menos documentación disponible para srsRAN.

En resumen, ambas opciones tienen sus propias fortalezas y debilidades, y la elección dependerá de las necesidades específicas de la organización. Se recomienda realizar una evaluación detallada antes de tomar una decisión final.

Como este en proyecto en general se trata de un proyecto de investigación para una asignatura de la universidad, se eligió srsRAN, que está diseñado para ser fácil de instalar y configurar, con una documentación detallada y una comunidad activa de desarrolladores y usuarios que ofrecen soporte y ayuda. Por otro lado, con las nuevas actualizaciones de OAI, han agregado nuevas funciones y características, lo que hace que la red sea más compleja y difícil de administrar. Esto puede requerir más tiempo y recursos, dos elementos de los que actualmente no se disponen.

Se procede a configurar el entorno.

## Como se ha dicho anteriormente, se partió de un ordenador completamente vacío, sin sistema operativo. Después de investigar, se decidió instalarse la última versión de Ubuntu, en concreto “Ubuntu 22.04.2 LTS”, que se puede encontrar en: <https://ubuntu.com/download/desktop> [ultima vez: 15/03/2023]

**Para poder instalar el sistema operativo se necesitó hacer uso de un pendrive [CARACTERÍSTICAS] ,** para grabar el ISO en el pincho, se montó previamente en el pincho UNetbootin, un software gratuito y de código abierto que se utiliza para crear unidades USB de arranque. Con UNetbootin, los usuarios pueden descargar una imagen de sistema operativo o una distribución de Linux desde Internet y crear una unidad USB de arranque desde la imagen. Esto permite a los usuarios instalar o probar un sistema operativo o distribución Linux sin tener que grabar un CD o DVD.

Una vez instalado y configurado el sistema operativo, el siguiente paso es instalar los driver, también conocidos como controladores, son programas que permiten que un dispositivo de hardware se comunique con el sistema operativo de una computadora y otros programas. En términos sencillos, los drivers son intermediarios entre el hardware y el software.

The bladeRF PPA es un Personal Package Archive (PPA) creado por Nuand, la empresa que fabrica y vende el dispositivo de radio definido por software (SDR) llamado bladeRF. El PPA contiene controladores y software para el bladeRF, así como otras herramientas y utilidades relacionadas con SDR.

El bladeRF es un dispositivo SDR que se utiliza para transmitir y recibir señales de radio en un amplio rango de frecuencias, desde 300 MHz hasta 3.8 GHz. El dispositivo se conecta a una computadora a través de USB y se controla mediante software. El software de bladeRF incluye herramientas para la configuración del dispositivo, la visualización de señales y la grabación de datos.

Al agregar el PPA de bladeRF a las fuentes de software de Ubuntu, los usuarios pueden instalar fácilmente los controladores y el software necesarios para utilizar el bladeRF en su sistema. Esto simplifica el proceso de instalación y configuración del bladeRF para los usuarios de Ubuntu, ya que no tienen que buscar y descargar manualmente los controladores y el software desde el sitio web de Nuand.

Para activar este PPA, se usan las siguientes líneas de código: (<https://github.com/Nuand/bladeRF/wiki/Getting-Started%3A-Linux>)

$ sudo add-apt-repository ppa:nuandllc/bladerf

$ sudo apt-get update

$ sudo apt-get install bladerf

Para comprobar que todo ha salido correctamente

bladeRF-cli -- -i

Ahora el entorno

Una vez instalados los drivers, instalaremos el software srsRAN. Para esto, se siguió los pasos que aparecen en [www.srslte.com](http://www.srslte.com).

Como no se planeaba usar una USRP, las líneas de comando usadas fueron:

Librerías necesarias

sudo apt-get install build-essential cmake libfftw3-dev libmbedtls-dev libboost-program-options-dev libconfig++-dev libsctp-dev

Descargar y construir srsRAN 4G:

git clone https://github.com/srsRAN/srsRAN\_4G.git

cd srsRAN\_4G

mkdir build

cd build

cmake ../

make

make test

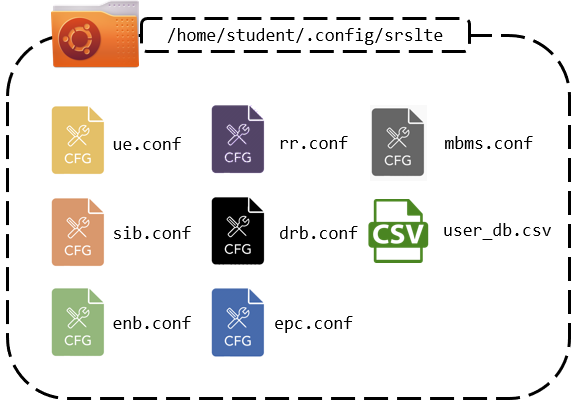
Instalar srsRAN 4G:

sudo make install

srsran\_4g\_install\_configs.sh user

## Una vez descargado el software, se comenzará a editar los ficheros de configuración.

## Los archivos de configuración de srsRAN son archivos que contienen información y parámetros de configuración para el software srsRAN. Estos archivos se utilizan para personalizar la configuración del sistema y adaptarla a las necesidades específicas de cada red o proyecto. Algunos de los archivos de configuración más comunes en srsRAN son:



En la figura anterior se pueden observar un total de 8 documentos: 7 de ellos corresponden a archivos de configuración y una base de datos (user\_db.csv). Esta base de datos se actualiza cada vez que un nuevo usuario se une a la red, y allí se almacena información como su IMSI, el algoritmo de autenticación utilizado, su clave, su QCI y la estrategia utilizada para asignarle una dirección IP.

A continuación, se resumen los principales aspectos que pueden ser adaptados en la red a través de la modificación de estos archivos de configuración:

* Configuración del UE: se utiliza el archivo ue.conf, que permite principalmente modificar los parámetros RF, de simulación del canal, la USIM, y parámetros físicos. Además, permite configurar los directorios donde se desean que se almacenen las capturas, trazas y logs obtenidos.
* Configuración del ENB: se dispone de 4 documentos para su adecuación, los cuales no son independientes entre sí. El archivo enb.conf permite cambiar los parámetros RF y la configuración de los logs, trazas y capturas, y también permite configurar la frecuencia de trabajo de la red y el número de PRB que va a asignar la red a los usuarios. El archivo sib.conf permite adecuar las características de los System Information Block (SIB). El archivo rr.conf se destina a la configuración de los recursos radio, y el archivo drb.conf está enfocado en permitir la configuración de los Data Radio Bearers (DRB).
* Configuración del EPC: se dispone del archivo epc.conf, que permite adaptar los parámetros del HSS, el MME, el túnel que comunica el SGW y el PGW. También se puede indicar en qué directorios se quieren guardar las trazas, las capturas y los logs obtenidos.
* Configuración del Multimedia Broadcast Multicast Service (MBMS): es un archivo centralizado en la configuración de la comunicación broadcast en la red desplegada. Sin embargo, para completar la configuración es necesario modificar los archivos ue.conf y enb.conf.

## En nuestro caso sólo nos hará falta editar los correspondientes al EPC y nodeB, ya que el UE es la tarjeta sim proporcionada.

## Configuración SRSENB