VIRTUAL EXPERIENCE
OCTOBER 11-14

2021 Fall Technical Forum
SCTE • NCTA • CABLELABS

Network in A Box with Open Source EPC/HSS and 'Zero Touch' Control

A Technical Paper prepared for SCTE by

Joerg Ahrweiler
Director Wireless R&D
Charter Communications
6360 S Fiddlers Green Cir, Greenwood Village, CO, 80111
+15613065021
Joerg.Ahrweiler@charter.com

Hany Heikal
Director Wireless R&D
Charter Communications
6360 S Fiddlers Green Cir, Greenwood Village, CO, 80111
+1720-595-4645
Hany.heikal@charter.com

Hossam Hmimy
Senior Director Wireless R&D
Charter Communications
6360 S Fiddlers Green Cir, Greenwood Village, CO, 80111
+17204049716
Hossam.hmimy@charter.com

Table of Contents

Title **Page Number**

1. Introduction 4
2. Cellular Network CBRS Architecture 4
   2.1. Radio Access Network (eNB) 4
   2.2. Core Network (EPC/HSS/PCRF) 5
3. End To End NIB integration and setup 5
   3.1. Physical connectivity 6
   3.2. Automated system bring-up and device monitoring – “Zero Touch” 7
      3.2.1. Introduction 7
      3.2.2. Presentation Layer 9
      3.2.3. Business Logic 13
      3.2.4. Data Layer 14
      3.2.5. Development Strategy 14
      3.2.6. Configuration Management 15
4. Mobile Edge Compute Use Case Examples 16
5. Conclusions 17

Abbreviations 18

Bibliography & References 18

List of Figures

Title **Page Number**

Figure 1: Network architecture 4
Figure 2: Fully-integrated NIB setup 6
Figure 3: Logical and physical connectivity 7
Figure 4: NIB ‘Zero Touch’ portal architecture 8
Figure 5: Initial status after power-up - next step wait for physical interfaces to come in service 9
Figure 6: All Ethernet interfaces are in service – next step start of EPC/HSS services 10
Figure 7: EPC/HSS services started – next step activation of S1 link 10
Figure 8: S1 link is activated – next step wait for positive SAS communication and eNB to come On-Air 11
Figure 9: The system is fully in service and eNB On-Air – user is notified that devices can be powered-on 12
Figure 10: After devices are powered-on, the device table is activated and device status monitored 12
Figure 11: The devices are successfully attached to the NIB network and the RF conditions reported and presented in the NIB portal 13
Figure 12: Corresponding device NIB application 13
Figure 13: eNodeB Simulator 15
Figure 14: Desktop Configuration 16
Figure 15: VR over NIB with low latency 17

List of Tables

Title **Page Number**

Table 1 – LTE eNB Equipment Specifications 5

1. Introduction

The Network In A Box (NIB) definition started with a 4G network that is operating the core network and base-station in a single box that is portable and self-organizing, which provides seamless connectivity to a group of mobile users, offering services such as internet connectivity and closed group communication (Push To Talk/Video/Text). The NIB setup expands cellular network coverage in various environments and different use cases, such as terrestrial disaster relief, private networks in-flight, at sea and in other scenarios and environments where an ad-hoc cellular network is required. The NIB concept is a miniature evolution of the traditional cell on the wheel (COW) concept that has been widely used for cellular mobile voice communication.

In this paper, the NIB cellular network architecture in a CBRS (Citizen Broadband Radio System) environment is described, and the background of combining an 'off the shelf' 4G eNB with a software-based open-source core network and a 'Zero Touch' system bring-up and control SW implementation will be presented. Finally, the testbed implementation of the 4G core network and eNB will be installed in a bare-metal environment with MEC (Mobile Edge Compute) demonstration application environment. The paper will conclude with lessons learned, and outlook to migrate to a 5G and container-based NIB.

2. Cellular Network CBRS Architecture

The network architecture in Figure 1 shows the traditional 4G network architecture with a CBRS environment. The RAN (Radio Access Network) and CN (Core Network) architecture in the NIB setup is very much like the traditional LTE network in commercial solutions; main difference is the small-scale nature of the setup. Since the use of the frequency band B48 (3550 MHz – 3700 MHz) is ideal for the NIB approach – shared spectrum/no spectrum needs to be owned – the connectivity of the NIB setup to a SAS (Spectrum Access System) is mandatory.

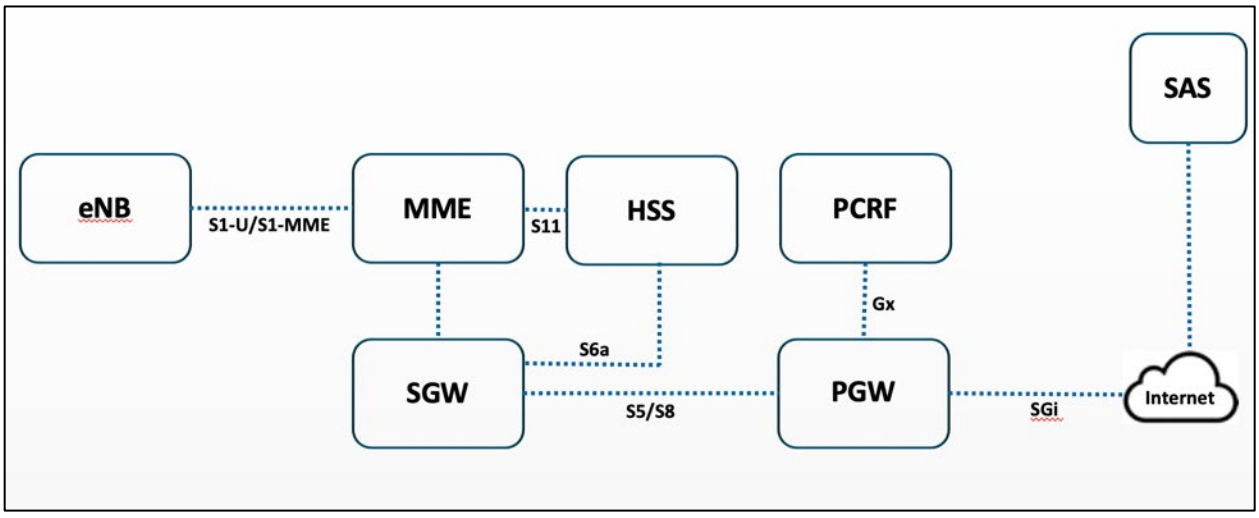

Figure 1 - Network architecture

2.1. Radio Access Network (eNB)

For the eNodeB, a commercial small cell Pico eNB is used. The output power was limited to 14 dBm/25mW per antenna port but can be increased to 24 dBm/250mW if required for respective use case.

Table 1 – LTE eNB Equipment Specifications

Specification	Value
Product	LTE Pico eNB
Band Support	B48 (3.55 – 3.7 GHz)
Carrier Aggregation	Up to 3 CA (only 1 Carrier used in NIB setup)
MIMO	2x2
Frame Configuration	TDD Frame Configuration 2 Special Sub Frame 7
IBW/OBW	150 MHz / 60 MHz
Output Power	2 x 10 W
Antenna	Built-in average 0 dBi
Modulation QAM DL/UL	256 / 64
BF Capability	No
CBRS Classification	CBSD CAT B

2.2. Core Network (EPC/HSS/PCRF)

The Core Network functions are established by utilizing an open-source application suite running on a bare-metal industrial small scale PC with a Linux-based operating system.

There are nowadays many options of open-source EPC/HSS solutions, see below for a list (not conclusive) of most popular open source core network:

- Open5Gs - Formerly NextEPC
- OpenAI Core Network - Related to / branched from OMEC
- Magma - Based on OMEC, with a focus on Fixed Wireless more than mobile
- OMEC – Open Evolved Mobile Core
- OpenMME – MME
- OpenCORD
- srsEPC

3. End To End NIB integration and setup

To be able to easily deploy the NIB, all the components are installed in a Pelican case for easy transportation. Additionally an embedded screen and wireless keyboard/mouse are included in the setup.

Figure 2 - Fully-integrated NIB setup

Figure 2 is showing the industrial type PC, running the EPC/HSS and O&M functionalities, on the left side in the case. The right component in the case is the commercial Pico eNB.

3.1. Physical connectivity

The eNodeB is physically interconnected to the industrial PC via 2 * RJ45 Ethernet cables. One connection is for the S1-MME&S1-MME LTE connectivity, the other one is for O&M administration purposes. The O&M link is utilized to automatically control the bring-up and maintain the system. A third Ethernet interface on the industrial PC is for the Backhaul connectivity to the Internet (user traffic and SAS connectivity). The same function can also be established via an internal WiFi or (Commercial) Cellular module. In the future, NIB versions and other BH connectivity options will be explored, e.g. MANET in unlicensed spectrum.

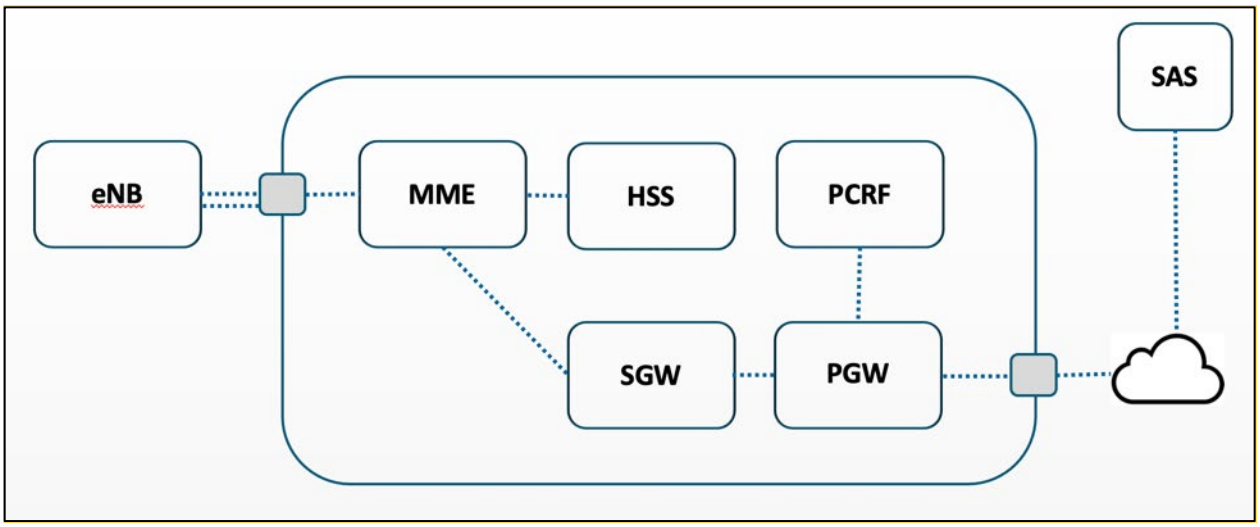

Figure 3 - Logical and physical connectivity

3.2. Automated system bring-up and device monitoring – "Zero Touch"

3.2.1. Introduction

For the purpose of a 'zero touch' and controlled system boot-up, a software application suite was developed to overcome any possible grace conditions and system malfunctions while the NIB setup is coming in service. The application starts automatically after power-up of the NIB and observing the physical and functional status of all components. If a delay or intervention during the boot-up process is necessary, the application will do so. Once the system in completely in service and the Radio is On-Air, the user will be notified and the power-on of the end-user devices can begin.

The requirement was to automate all manual processes to start the private wireless network, known as Network in a Box. Therefore, the start mechanism has to follow the new concept of Zero Touch.

In order to implement a Zero-Touch start mechanism for the Network in a Box, a three-tier software solution has to be developed.

The most common software architecture for typical client-server applications is three-tier architecture, which divides applications into three logical and physical computer tiers, as shown in Figure 4.

The presentation tier, or user interface; the application tier, where data is processed; and the data tier, where the data associated with the application is stored and managed, is a well-established software application architecture that organizes applications into three logical and physical computing tiers.

For many years, the three-tier design was the standard for client-server applications. Most three-tier programs are now candidates for modernization and cloud migration using cloud-native technologies like containers and microservices.

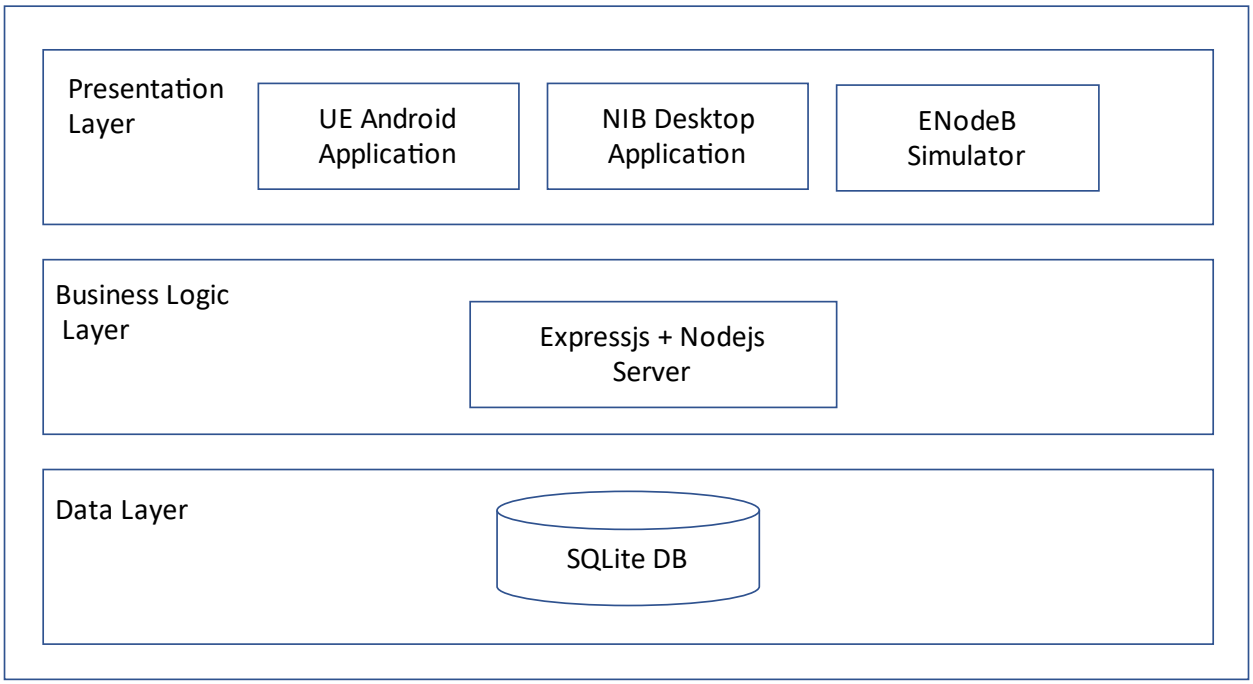

Figure 4 - NIB 'Zero Touch' portal architecture

3.2.2. Presentation Layer

The idea here is to have two applications, the first one is a desktop application, and the second one is an android application.

The desktop application runs on the ubuntu server, and it starts automatically once the server starts. The desktop application automatically starts the private LTE components (MME, SGW, PGW, PCRF, and HSS). It checks the log file for all of these components to ensure that each component starts correctly. It displays a block diagram to represent the elements of the network and it uses the color scheme to represent the status of each element as shown in the screenshots below.

The software used to build this layer are Reactjs and Electronjs and can be located in referenced links 1 and 2.

The desktop application has to communicate with the operating system and it executes a number of the commands in the command-level. In order to do that, the JavaScript child_process package had to be used.

The child_process module creates new child processes of our primary Node.js process. We can execute shell commands with these child processes.

Using external processes can improve the performance of an application if used correctly. For example, if a feature of a Node.js application is CPU intensive, as Node.js is single-threaded, it would block the other tasks from executing while it is running.

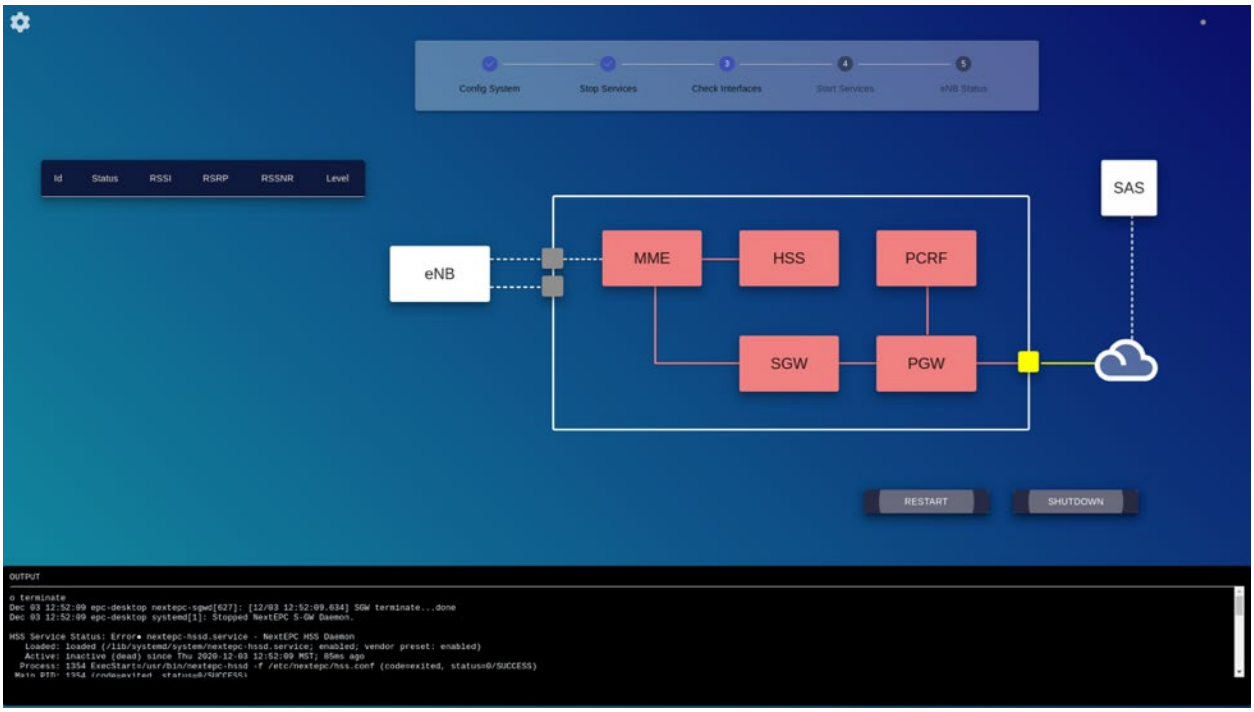

Figure 5 - Initial status after power-up - next step wait for physical interfaces to come in service

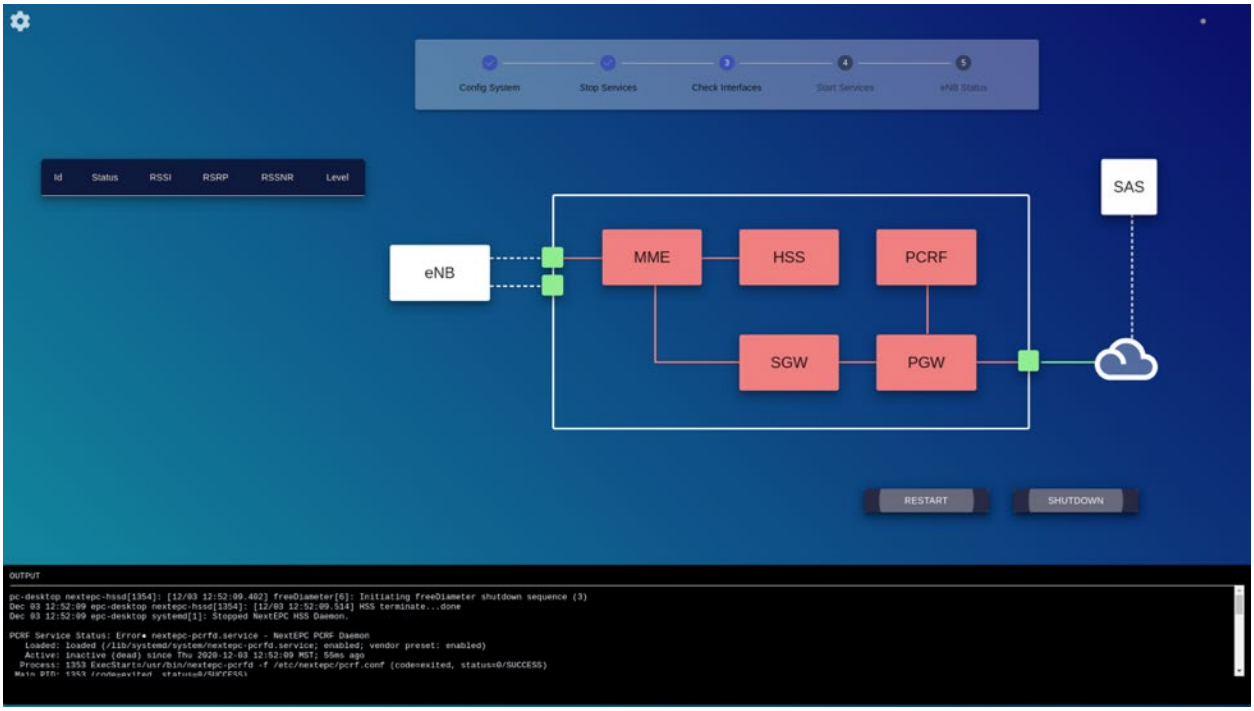

Figure 6 - All Ethernet interfaces are in service – next step start of EPC/HSS services

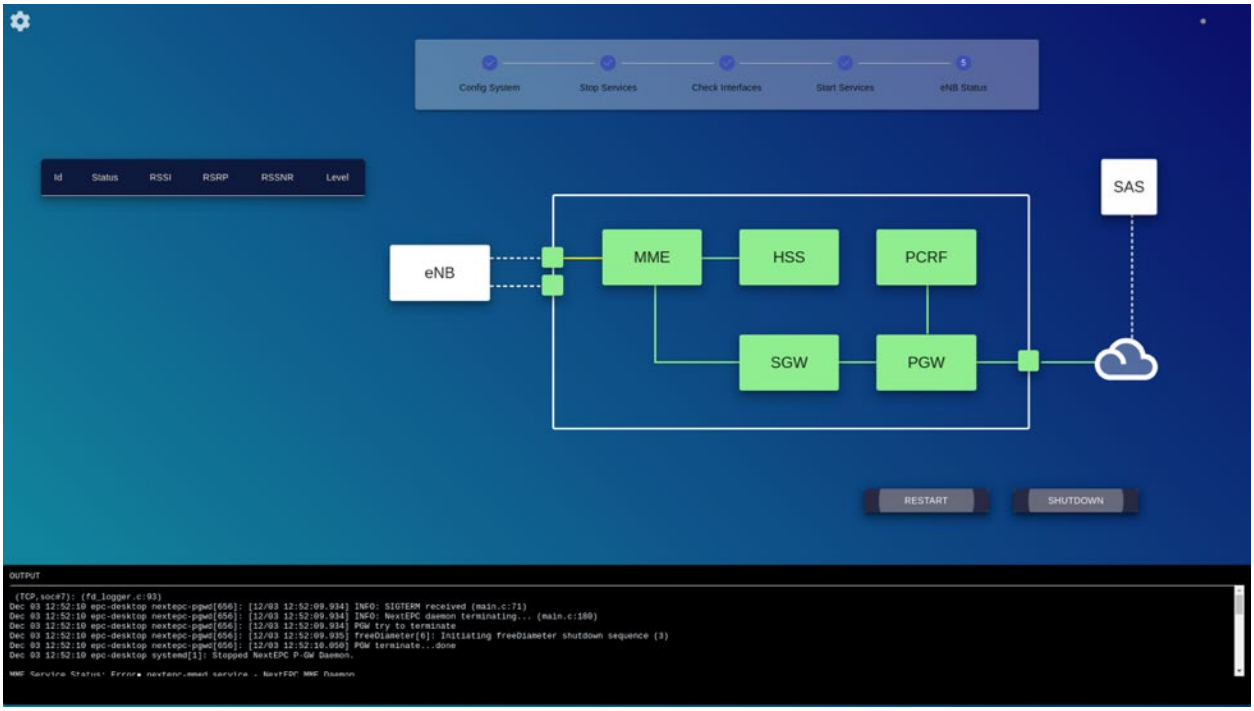

Figure 7 - EPC/HSS services started – next step activation of S1 link

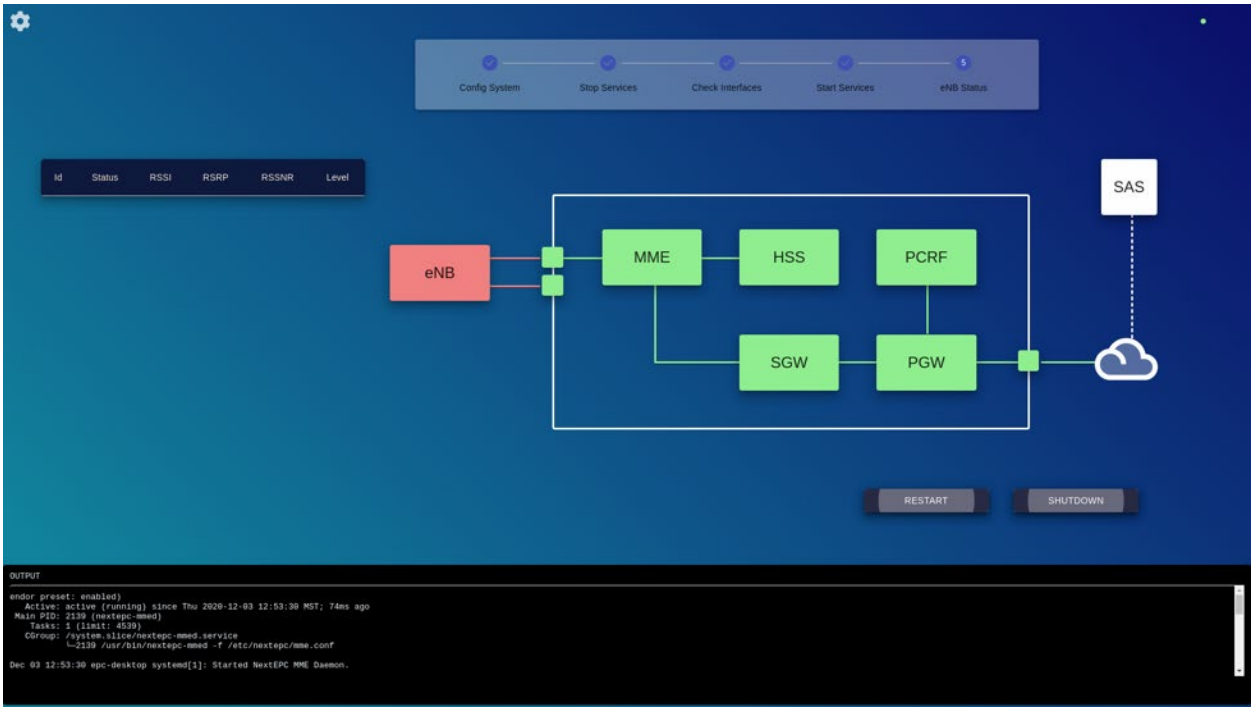

Figure 8 - S1 link is activated – next step wait for positive SAS communication and eNB to come On-Air

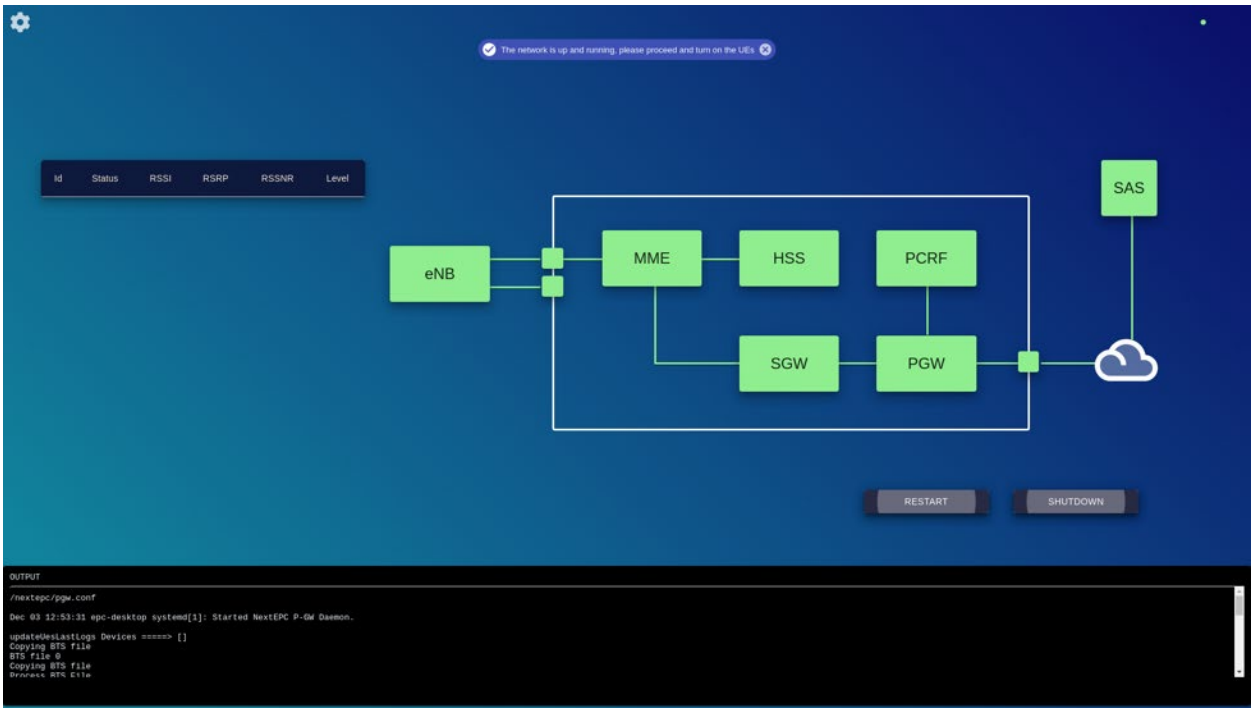

Figure 9 - The system is fully in service and eNB On-Air – user is notified that devices can be powered-on

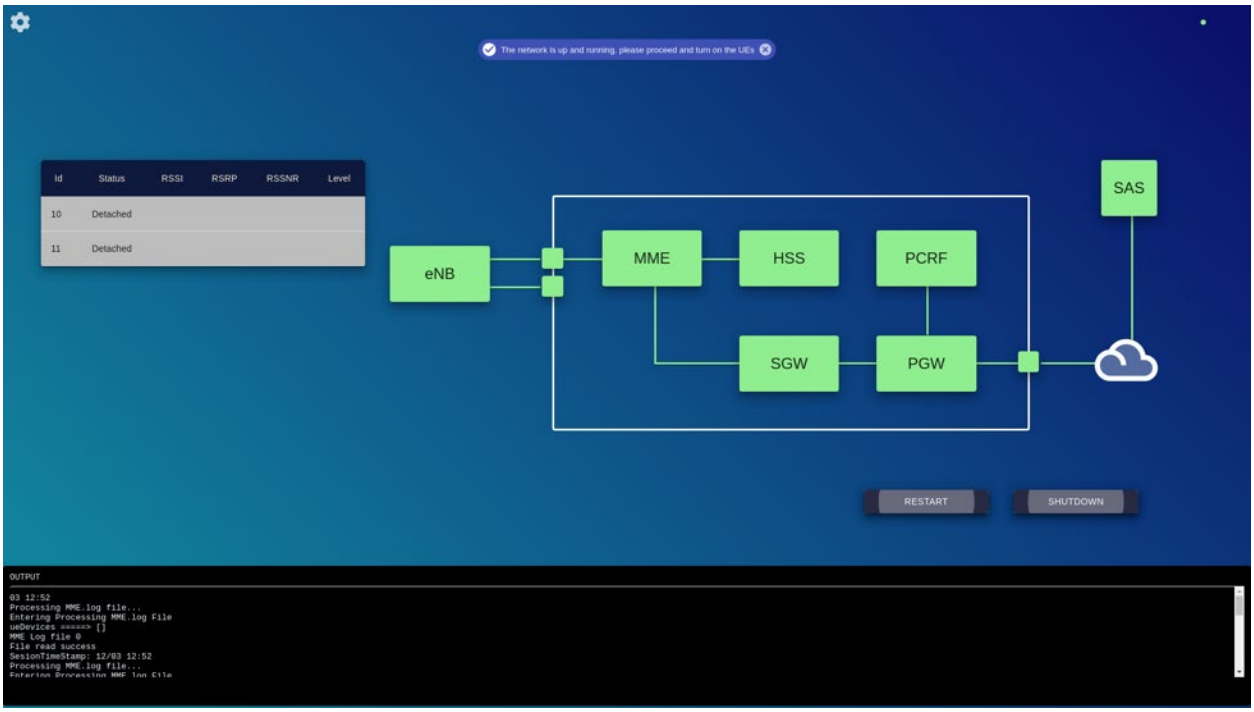

Figure 10 - After devices are powered-on, the device table is activated and device status monitored

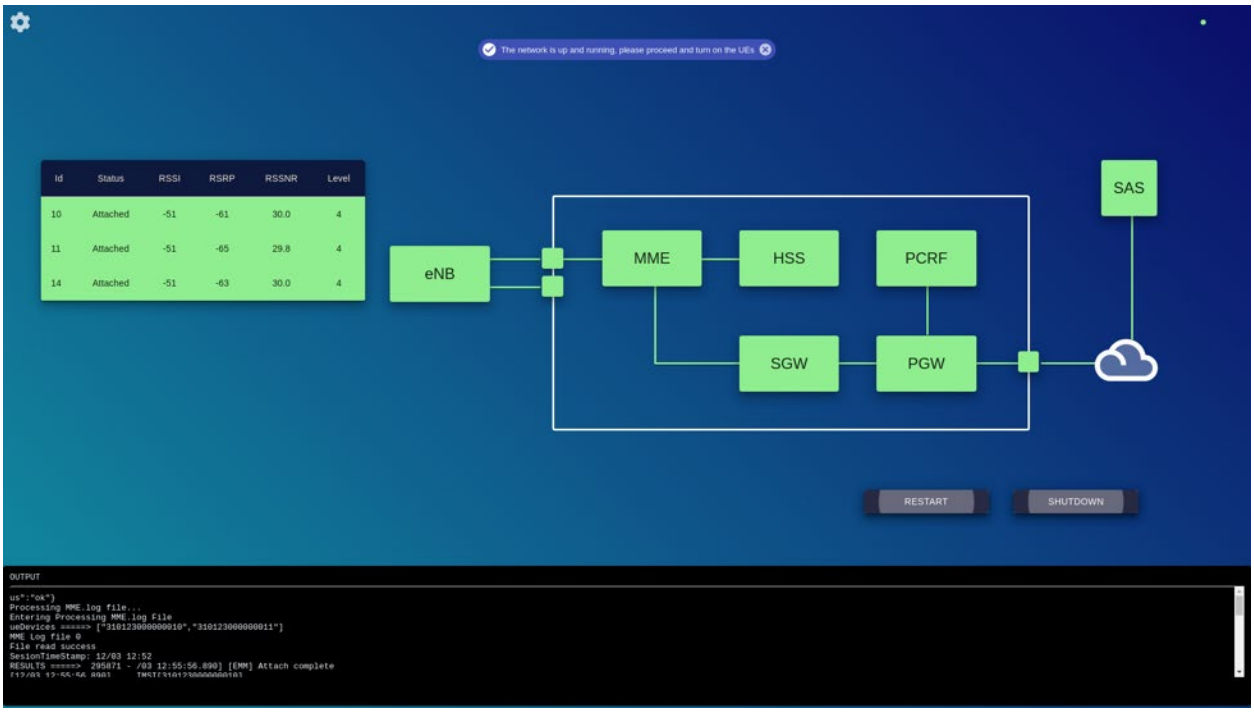

Figure 11 - The devices are successfully attached to the NIB network and the RF conditions reported and presented in the NIB portal

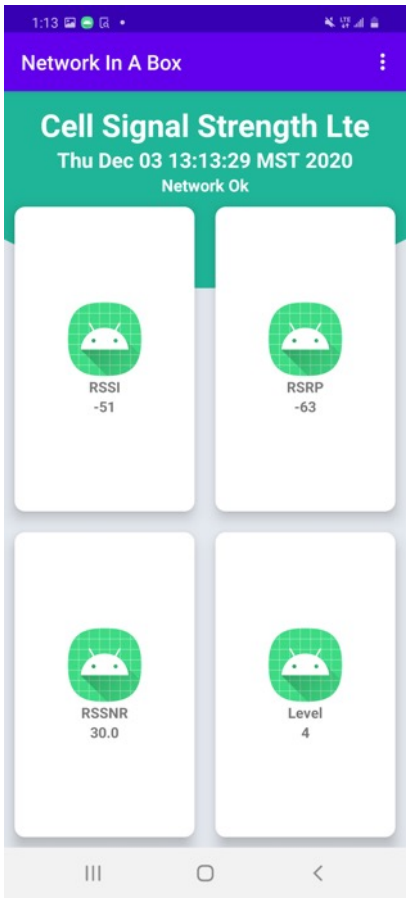

Figure 12 - Corresponding device NIB application

3.2.3. Business Logic

Three software components had to be used in this layer. The software list is ExpressJS, NodeJS and Puppetee,r and can be located into the following referenced links 3-5.

The ExpressJS was used to build a Rest API interface to serve all the HTTP requests from the Desktop Application.

Express is a Node.js web application framework that offers a comprehensive range of functionality for both web and mobile apps.

Express adds a thin layer of basic web application functionality without obscuring the Node.js capabilities users already know and appreciate.

Node.js is a scalable network application builder that uses an asynchronous event-driven JavaScript engine. Many connections can be handled at the same time in the "hello world" example below. The callback is invoked with each connection, but if there is no work to be done, Node.js will sleep.

Puppeteer is a Node library that provides a high-level API to control headless Chrome or Chromium over the DevTools Protocol. It can also be configured to use full (non-headless) Chrome or Chromium.

A headless browser is a great tool for automated testing and server environments where one does not need a visible UI shell. For example, one can run some tests against an actual web page, create a PDF of it, or inspect how the browser renders an URL.

The Chrome DevTools Protocol allows for tools to instrument, inspect, debug and profile Chromium.

3.2.4. Data Layer

The database which was used to store the data is SQLite and it can be found in the referenced link 6.

The SQLite file format is stable, cross-platform, and backwards compatible
The following information is stored in the database:

UeID: User Entity Identification
RSSI: Received Signal Strength Indicator
RSRP: Reference Signal Received Power
RSSNR: Reference Signal Signal to Noise Ratio
CQI: Channel Quality Indicator
level: Battery Level
timeStamp: Time Stamp

3.2.5. Development Strategy

It was not possible to have an environment for each developer who designs with the application.

The strategy was to create a simulator for the ENodeB as shown in Figure 13.
The ENodeB simulator was built using the ReactJS and NodeJS in the referenced links 1-2.

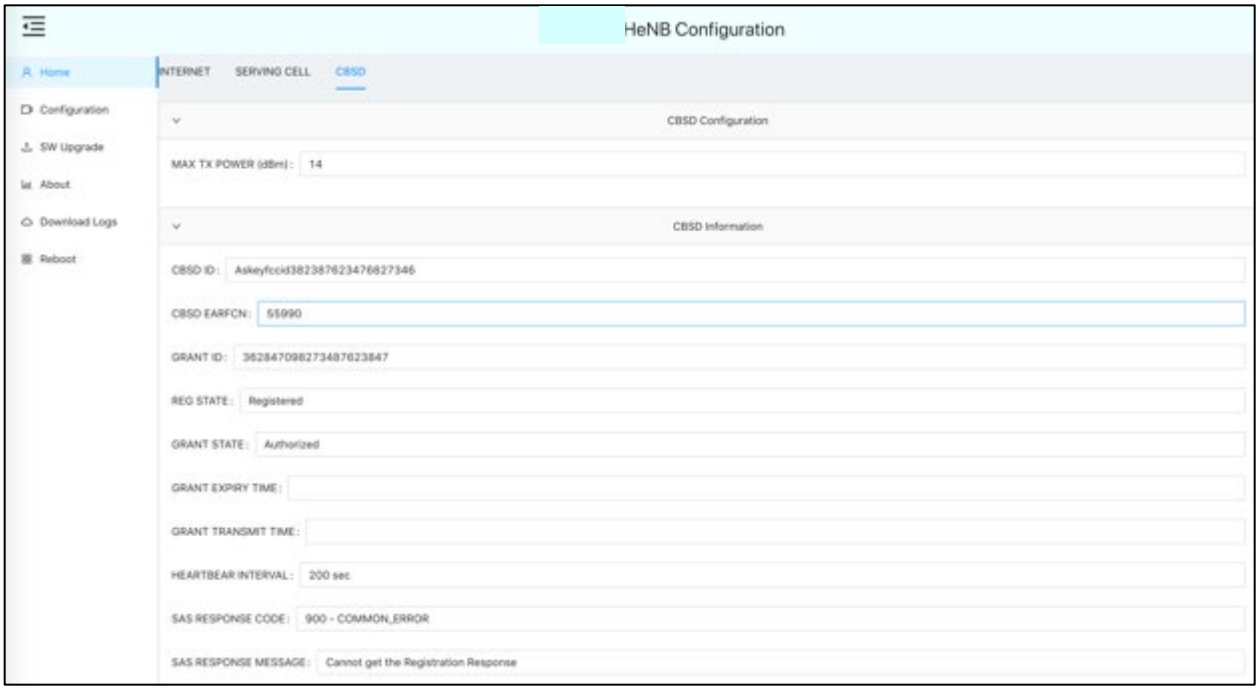

Figure 13 - eNodeB Simulator

3.2.6. Configuration Management

A Configuration had to be added for the user to be able to choose between which ENodeB is connected to the PLTE Core Network, as shown in Figure 14.

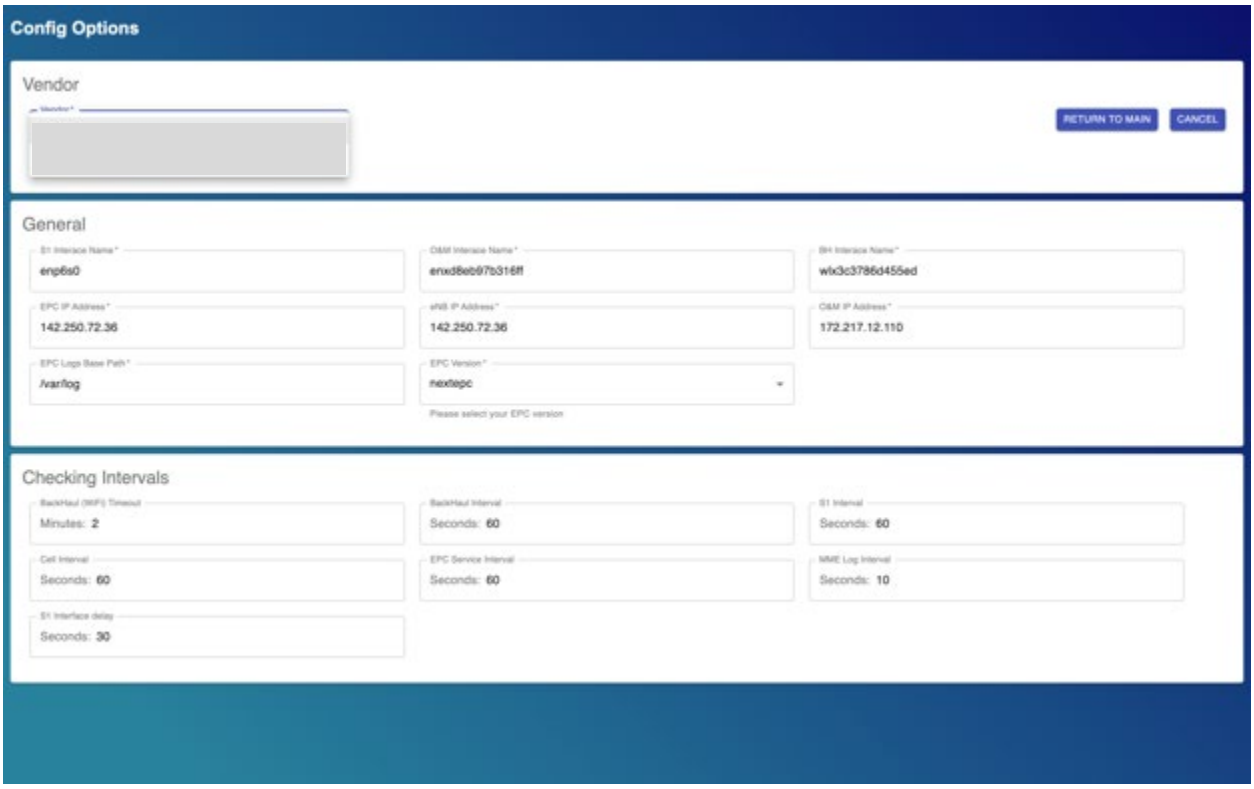

Figure 14 - Desktop Configuration

4. Mobile Edge Compute Use Case Examples

Beside generic end-user internet access, different options for MEC-based use cases were explored. The following lists two examples:

Closed Group Communication

Closed Group Communication aka 'Push To Talk/Video/Text' is an ideal communication method in an enterprise private wireless environment, e.g. factory, hotel etc... End users can be easily communicating with their respective groups (all or some at once or one-to-one), or a local PTx network dispatcher, by a push of a button. In our NIB solution, we installed a demo version the PTx server and dispatch applications in the MEC and respective PTx client application on the end-user devices.

VR (Virtual Reality) low latency applications

VR Workplace: Teleport to virtual office floor, Interact virtually with lab - machines and sensors

VR smart light control: Able the city workers to access Smart light for control and data retrieval using VR

VR Healthcare: Control instruments remotely, read patient charts

VR Industrial: Troubleshoot and analyze machine performance

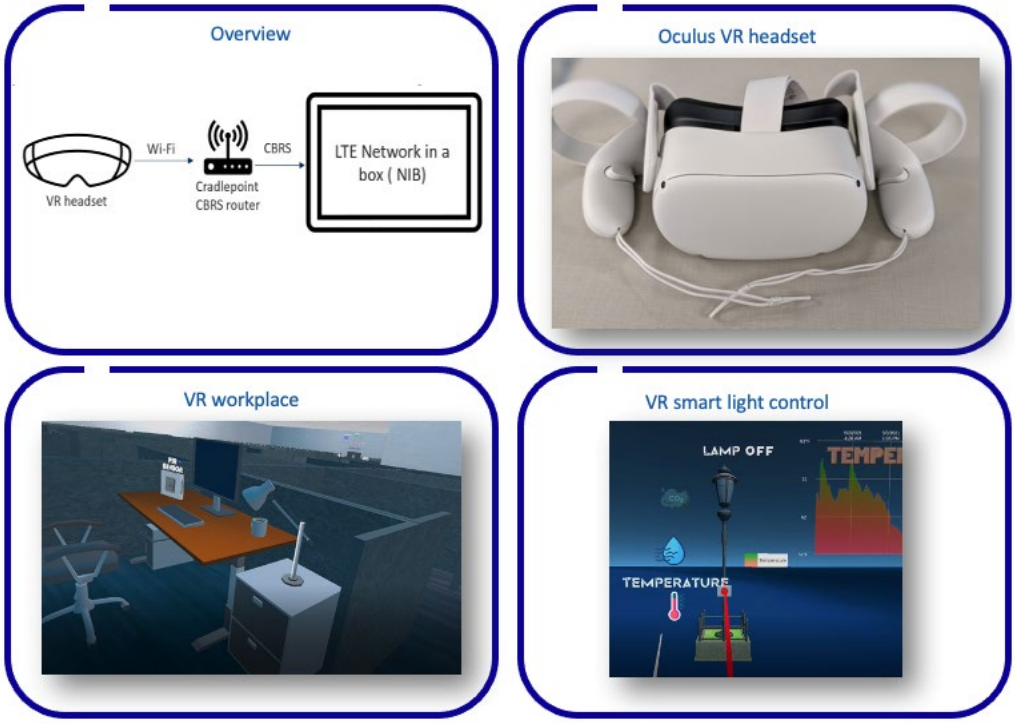

Figure 15 - VR over NIB with low latency

5. Conclusions

Network In A Box has a well-deserved reputation in the cellular industry. With the addition of 'zero touch' control and MEC in the same compute platform, the use for any users and many applications domains is opened up.

- Research and further development of NIB will continue which includes:
  - Evolve to a 5G system, e.g. 5G n48 Pico eNB and a 5G SA Core Network
  - Expand on the list of supported RAN vendors in the NIB 'zero touch' portal
  - Migrate the CN to a virtualized/containerized implementation

Abbreviations

CBRS	Citizen Broadband Radio System
CN	Core Network
COW	cell on the wheel
MEC	Mobile Edge Compute
NIB	Network In A Box
RAN	Radio Access Network
SAS	Spectrum Access System
VR	virtual reality

Bibliography & References

1- https://reactjs.org/

2- https://www.electronjs.org/

3- https://expressjs.com/

4- https://nodejs.org/en/

5- https://developers.google.com/web/tools/puppeteer

6- https://www.sqlite.org/index.html

