International Journal of Computing and Digital Systems
2025, VOL. 17, NO. 1, 1-14

http://dx.doi.org/10.12785/ijcds/1570990891

NomadicBTS-2: A Network-in-a-Box with Software-Defined Radio and Web-Based App for Multiband Cellular Communication

Emmanuel Adetiba[1,2], Petrina C. Uzoatuegwu[1], Ayodele H. Ifijeh[1], Abdultaofeek Abayomi[3], Obiseye Obiyemi[4,5], Emmanuel O. Owolabi[6], Katleho Moloi[5], Surendra Thakur[7] and Sibusiso Moyo[8]

[1]*Department of Electrical and Information Engineering Covenant Applied Informatics and Communication African Center of Excellence, Covenant University, Ota 1023, Ogun State, Nigeria*
[2]*HRA, Institute for Systems Science, Durban University of Technology, Durban, 4001, South Africa*
[3]*HRA, Walter Sisulu University of Technology, East London 5200, South Africa*
[4]*Space Science Centre, Durban University of Technology, Durban, 4001, South Africa*
[5]*Department of Electrical Power Engineering, Durban University of Technology, Durban, 4001, South Africa*
[6]*Smartyou Integral (PTY) LTD, 405, Justice Mahomed St., Pretoria, Gauteng, 0002, South Africa*
[7]*KZN e-Skills CoLab, Durban University of Technology, Durban, 4001, South Africa*
[8]*School for Data Science and Computational Thinking and Department of Mathematical Sciences, Stellenbosch University, Stellenbosch, 7602, South Africa*

Received 24 January 2024, Revised 14 September 2024, Accepted 19 September 2024

Abstract: The proliferation of mobile communications technologies has greatly contributed to the development of emerging economies. However, a significant digital divide remains in many remote and hard-to-reach regions due to the high Capital Expenditure (CapEx) and Operating Expenditure (OPEX) of mobile network operators. This study addresses the issue by developing and prototyping a cost-effective, software-defined base station, called NomadicBTS-2, utilizing open-source technologies and the Software-Defined Radio (SDR) paradigm. NomadicBTS-2 incorporates the Universal Software Radio Peripheral (USRP) B200 as the Radio Frequency (RF) hardware front-end. The software backend is built on open-source solutions, including the USRP Hardware Driver (UHD) and services such as OpenBTS, Asterisk, SIPAuthserve, and SMQueue. Additionally, a custom web-based software, NomadicBTS WebApp, was developed to configure and monitor the UHD and software services via a Graphical User Interface (GUI). The system was tested using two mobile stations (MSs) for simplex and duplex communication, while network link quality parameters were evaluated to assess users' Quality of Experience (QoE). Experimental results demonstrated that, within a pico-cell environment, the link quality is sufficient for call routing and Short Messaging Services (SMS) between user-to-user and network-to-user communications. NomadicBTS-2 presents a robust solution for a Network-in-a-Box, ideal for short-range communication in rural and hard-to-reach areas, emergency response scenarios, and IoT sensor networks. It can also be deployed to alleviate network congestion in existing mobile networks. Furthermore, this prototype offers a viable testbed for teaching and research laboratories to explore advancements in SDR, cognitive radio, and other wireless communication technologies, thus opening new frontiers in wireless communications research.

Keywords: NomadicBTS-2, Software-Defined Radio, Multiband Cellular Communication, Network-in-a-Box

1. Introduction

The emergence of wireless communication technologies has witnessed significant development over the past few decades [1], [2], [3]. High-speed heterogeneous networks have transcended from 2G, 3G, and 4G to 5G and beyond. However, the architecture for radio access technologies consists of the access layer, distribution/backhaul, and core network layer. Base Transceiver Stations (BTSs), which transmit and receive signals in the form of electromagnetic waves, make up the access network for users to interface with the network services and resources. The upsurge in CapEx and OPEX of mobile network operators inspires leveraging the Software Defined Radio (SDR) technology to achieve efficient, cost-effective, and most importantly, dynamic wireless communications. Aside from the numerous benefits of implementing SDR technology [4], an SDR can be positioned at the center of a pico/femto/microcell, that is, the smallest geographical area of a network, to provide an alternative to the traditional BTS [5]. It can also be useful in disaster-affected areas where telecommunications

E-mail address: emmanuel.adetiba@covenantuniversity.edu.ng, petrina.uzoatuegwu@stu.cu.edu.ng, ayodele.ifijeh@covenantuniversity.edu.ng, taofeekab@yahoo.com, obiseyeo@dut.ac.za, owolabi.emmanuel@gmail.com, katlehom@dut.ac.za, thakur@dut.ac.za, smoyo@sun.ac.za

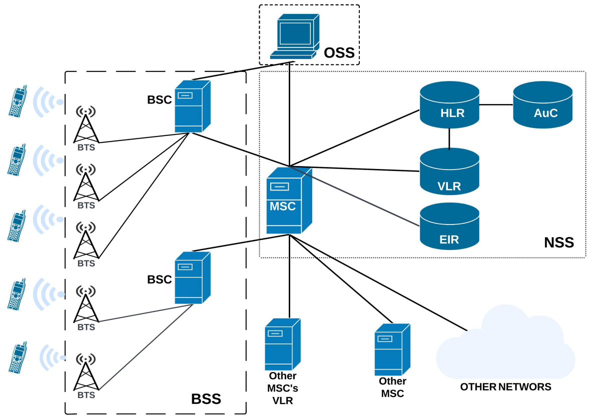


Figure 1. Traditional GSM Architecture

infrastructure is in dilapidation and to provide short-range communication in rural areas. The High Altitude platforms (HAPs) is another system that can be easily deployed in disaster zones (e.g earthquake, flood, hurricane etc.) [6], rural or non-network-available areas; however, HAPs might not be as cost-effective as NormadicBTS-2.

The ubiquitous role of mobile devices and the penetration of high-speed internet technologies are the key drivers to achieving a sustainable economy. As more users subscribe to the networks daily, the demand for improved voice and text messaging services continues to rise, leading to resource management and scalability concerns for mobile network operators. The Global System for Mobile Communication (GSM), a 2G technology, provides the underlying architecture for circuit-switched voice and SMS services [7]. It is known as the most popular wireless technology because of its widespread use and global penetration.

As shown in Figure 1, the GSM architecture is made up of three core parts: the Base Station Subsystem (BSS), the network switching subsystem (NSS), and the operational support subsystem (OSS). The BSS is known as the access network and consists of several BTSs, Base Station Controllers (BSCs), and Mobile Stations (MSs) interfacing with the BTSs. The NSS controls the switching to a home public network and serves as a gateway to other mobile networks, while the OSS gives support for the operation and maintenance of the GSM network, and also permits monitoring of various nodes of the GSM network. The BSS has a primary function of providing and managing radio transmissions between the MS and Mobile Switching Center (MSC), which routes traffic (voice, data, and SMS) to and from the GSM network. The BSS comprises two nodes, the BTS and BSC, with a radio interface (Abis) between them [8].

In a GSM network, the BTS performs over-the-air-interface radio transmission to and from mobile stations, while the BSC interconnects tens to hundreds of BTSs, coordinates frequency and channel allocation, and handover between cells. The MSC is responsible for radio link management, routing and switching, and communication within the Public Land Mobile Network (PLMN). Other functions of the BTS include modulation and demodulation, encryption, ciphering, channel coding, and decoding [8]. The intricate characteristics of the traditional BTS motivated the development of the SDR technology [9], [10]. It is a modern radio communication technology in which components that were traditionally implemented on hardware are now implemented by means of software on an embedded system or personal computer [11], [12]. SDR technology helps to mitigate the high CAPEX/OPEX challenges of fixed infrastructures and facilitates cross-functionality, cost-effectiveness, flexibility, and portability to drive reliable communications through to end users [5], [9], [13].

The application and use of SDR frameworks in the provisioning of wireless communication services have created several research motivations. To enable communication in remote areas, Wibisono et al. [14] designed and developed a “flying BTS” using a Raspberry Pi microcontroller as a web server, a Wi-Fi dongle to serve as an access port, a 4W 2.4 GHz amplifier to amplify the signal levels, and a 2.4 GHz antenna as the radiator, all packed in a container and placed in a balloon. Hafidudin et al. [15] used the OpenBTS and USRP to provide alternative access to local mobile operators. The OpenBTS running on a Linux OS was connected to a Voice over IP (VoIP) service to form a quadruple play service and Asterisk was used to connect with other telephone networks.

[16] employed SDR technology for monitoring the electromagnetic field using the implementation of a cost-efficient ADALM-PLUTO SDR platform. This configuration employs a Raspberry Pi 4 Model B, which is equipped with a Broadcom 2711 CPU. The processor has a 64-bit Cortex A72 architecture, operates at a frequency of 1.5 GHz, and consists of 4 cores. Additionally, it is accompanied by 4 GB of LPDDR4 SDRAM. The platform was only utilised for the observation of signals sent by the Terrestrial Trunked Radio (TETRA) communication standards. The implementation of this system yielded satisfactory performance, along with exceptional mobility and flexibility, rendering it well-suited for outdoor settings. Similarly, [17] examined the possibility of using two inexpensive software-defined radios, Adalm-Pluto and LimeSDR USB, to replace expensive metrology equipment, with a particular reference to one of the prominent brands. The purpose of the research was to identify the unique metrological properties of these SDRs and then compare them to the more conventional metrology tools. With a spectral density ratio similar to the SFE100 and uncertainties of 0.5 dBm, the findings showed that SDRs could provide test signals (DAB, ISDB, FM, DVB, and SXM), thus suggesting that SDRs can successfully replace that particular metrological equipment brand.

Using algorithms on SDR platforms, the DronEnd research project in [18] developed and deployed a drone detection and defence system [19]. To prevent unauthorised drones from entering a designated region, the DronEnd ground defence system scans the radio frequency spectrum, detects their existence, uses Angle of Arrival (AoA) algorithms to pinpoint their location, and then uses radio frequency jamming techniques to disable them. To detect drone transmissions, the radio spectrum must be monitored by RF-based drone defense systems. The DronEnd system made use of energy detection-based spectrum sensing methods. The USRP X310 with Twin-RX RF Daughterboards is the USRP SDR platform used on which algorithms like 3EED and 3EED with an adjustable threshold were implemented, which span a frequency range of 10-6000 MHz. Since the SDR platforms can receive signals up to 6 GHz, the system was tested on drones utilizing the 2400-2500 MHz and 5730-5830 MHz ISM bands.

In a similar study, Muslimin et al. [20] simulated the effects of the modulation process using a setup comprising of the M-PSK and M-QAM modulation schemes, in addition to the GNU Radio v3.7.8.2. running on a computer system. For the real node to transmit and capture real signals, multiple USRP N210 was used with the CBX (40 MHz) daughterboard linked to a host computer at both the receiver and transmitter sides and a loopback cable as the communication channel of the USRPs. For efficient communication in remote areas, Prasannan et al. [21] proposed a solution through an OpenBTS-based microtelecom model following the SDR paradigm using a USRP. Further testing and observation proved that using a small microcontroller like the Raspberry Pi and open-source technologies such as the YateBTS is cost-effective and mobile. Similar to [21], Aggrawal and Vachhani [22] offered an alternatively cost-effective procedure for immediate restoration of remote connectivity in the event of disasters; a BTS was built using the USRP B200 board and OpenBTS based on the SDR technology. Other related works are summarised in Table I.

In [23] OpenBTS and OpenAirInterface were used to compare GPRS and LTE network performance, highlighting LTE's quality optimisation. [22] employs USRP B200 and OpenBTS to restore wireless connectivity in disaster-hit areas by establishing a cellular GSM network at a fraction of the cost and time of regular infrastructure, focusing on localised voice and SMS services. [15] discusses OpenBTS's use of USRP for quadruple play services, its benefits over GSM networks, and its successful implementation for VoIP services meeting QoS standards in areas without local mobile operators. [24] presents the vulnerability assessment of a GSM-based IoT M2M node, highlighting the potential of SDR and OpenBTS for penetration testing and vulnerability analysis.

In [25] OpenBTS with USRP N210 in use to create a simple, inexpensive, and portable GSM telecommunications infrastructure for remote areas using SDR technology to overcome the challenges of traditional GSM network implementation in low-populated or geographically isolated areas. [26] presents the development of OpenBTS, a mobile analysis system for natural disaster victim search, emphasising disaster awareness, community preparedness, cell phone victim detection, and the need for an emergency GSM network. [27] describes how OpenBTS was used to implement low-cost mobile operator solutions for prepaid voice services due to its cost-effectiveness, flexibility, and successful billing system integration.

A GPRS network prototype using SDR and OpenBTS for IoT application testing is presented in [28], emphasising the importance of GPRS in remote areas and SDR for prototyping. [29] describes a USRP and OpenBTS GSM network that can perform voice, SMS, MMS, and GPRS, as well as its benefits and use cases, while [30] describes telecommunications engineering in mobile networks, student practical experience, and educational implementation of 2G, 4G, and 5G NSA systems using OpenBTS and OpenAirInterface.

Our work in [11], which is a precursor and the first version of the current work is a NomadicBTS, based on SDR technology that was designed, built, and prototyped to provide an easy-to-deploy and economical communication solution. The design consists of an RF frontend and a software backend. The RF frontend is the Ettus Research USRP B200 which has a frequency range of 50 MHz to 6 GHz [31]. Results from the tests carried out within the Covenant University campus showed that the appropriate cell radius of the prototype was 7.54 m. Further tests with two test phones showed that the MSs could access the network for voice and SMS with minimal delay on channel assignment. However, the configuration of the NomadicBTS involves the use of Ubuntu OS Command Line Interface (CLI), which is not efficient in practice. Also, the work also considered only one single frequency band in the 900MHz spectrum while other higher bands for GSM technology were not explored. These are the gaps being addressed in the current paper.

The summary of the main contributions in NomadicBTS-2 is highlighted as follows:

- Exploration of multiple GSM spectrum bands.
- Utilization of SDR (Software-Defined Radio) hardware and frameworks.
- Integration of open services including USRP B200, OpenBTS, Asterisks, SIPAuthserve, SMQueue, and NodeManager.
- Development of NomadicBTS WebApp for GUI-based configuration and monitoring of open-source software services.

TABLE I. Other Related Works

Reference	Applications used	USRP type
[23]	OpenBTS, OpenAirInterface, Linux	USRP B210
[22]	OpenBTS, Asterisk, SIPAuthServe, SMQueue, USRP Hardware Driver (UHD)	USRP B200
[15]	GNU Radio, Open BTS, IMS Core, Asterisk software	USRP B205
[24]	OpenBTS, Python	USRP N210
[25]	GNU Radio, Universal Software Radio Peripheral (USRP), UHD driver	USRP N210
[26]	PyBOMBS, Asterisk, Smqueue, Crontab	USRP RAD1
[27]	OpenBTS, Asterisk PBX, UHD, Sipauthserve, smqueue, LabView, GNU Radio	USRP N210
[28]	OpenBTS, USRP, GNU Radio	USRP B210
[29]	OpenBTS, Asterisk, USRP, GNU Radio, UHD	USRP B210
[30]	OpenBTS, OpenAirInterface SDR modules	USRP B210

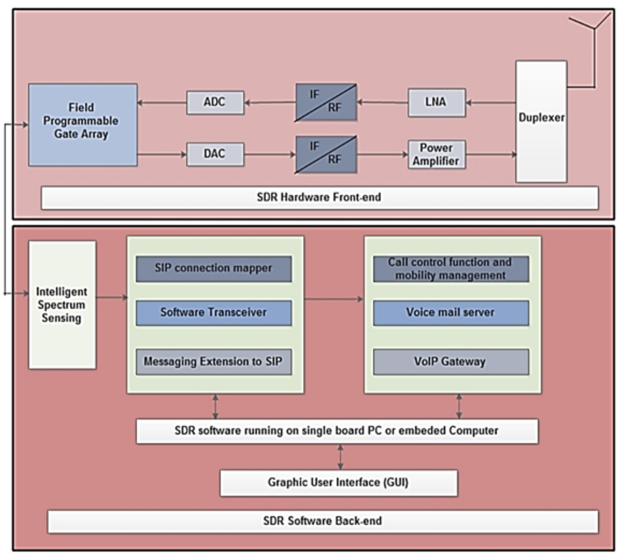


Figure 2. Architecture of NomadicBTS-2

- Implementation of a billing service for controlling, monitoring, recording, and charging customer calls and text messages over the NomadicBTS network.

The rest of this paper comprises of the materials and methods in Section II, results, and discussion in Section III, and finally, the conclusion is presented in Section IV.

2. Materials and Methods

A. System Architecture

NomadicBTS-2 is based on the SDR architecture as shown in Figure 2. It comprises the hardware front-end (USRP B200), which is responsible for over-the-air communication between the MS and the BTS, and the software backend, which is installed on a general-purpose computer. The NomadicBTS-2 supports multiple frequencies in the GSM band (e.g., 1800 MHz and 1900 MHz), and is used for the provisioning of voice and SMS services. Notably, the selection of OpenBTS over OsmoBTS and YateBTS is subject to its compatibility with a large number of SDRs.

B. Hardware Frontend

The hardware front end consists of the USRP B200 with the VERT900 antenna. The USRP B200 is a small-size Single Input Single Output (SISO) SDR radio. It has a frequency range of 50 MHz to 6 GHz and a Xilinx Spartan-6 XC6SLX150 Field Programmable Gate Array (FPGA). The RF part makes use of an AD9364 RFIC transceiver that has 56 MHz of contiguous bandwidth and is connected via a USB 3.0 port for transferring data to and from the computer. At the receiver sub-block, the VERT900 antenna receives the signal and forwards it to a Low Noise Amplifier (LNA) through the duplexer. The RF signal is then converted to a corresponding intermediate frequency (IF), which can be digitised by the Analog to Digital Converter (ADC).

The FPGA handles signal decoding, data rate conversion, and timing. At the transmitter sub-block, the processed signal from the FPGA is converted back to analog waveform by the Digital to Analog Converter (DAC), which is thereafter converted from IF to RF. The amplitude of the RF signal is then boosted by the power amplifier, and ultimately transmitted to the antenna through the duplexer. The primary function of the duplexer is to multiplex signals to and from the MSs into the USRP B200. The VERT900 antenna is an 824 MHz to 960 MHz and 1710 MHz to 1990 MHz quad-band cellular omnidirectional antenna with 3 dBi gain [31]. It is 8 inches in length and features a tilt-and-swivel SMA-Male connector, which makes it suitable for use vertically, at a right angle, or any angle in between. The SMA-Male connector also makes it compatible for use with the USRP B200 device.

C. Software Backend

The software backend is made up of the OpenBTS and its dependencies, such as the SIPAuthServe, SMQueue, and Asterisk, which are all installed on the Ubuntu OS. OpenBTS implements a complete GSM stack for voice and SMS as well as supporting GPRS and UMTS (3G) data standards. It permits the radio-air interface to interconnect with other nodes on the GSM network [32]. It performs various functions of a BSC and BTS independently or in tandem with other software for cost-effectiveness [33]. Asterisk is a VoIP switch that handles the Session Initiation

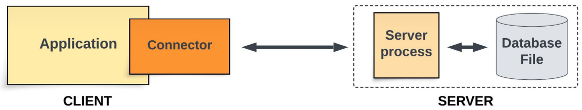


Figure 3. SQLite Block Diagram for Database Access [32]

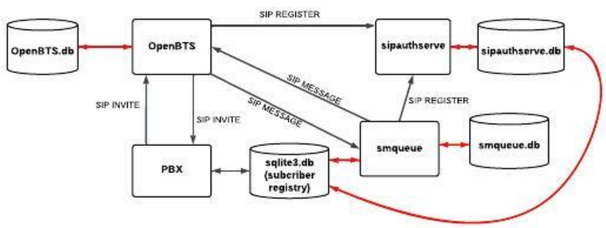


Figure 4. Interconnection of the Backend Open-Source Services

Protocol (SIP) INVITE requests, establishing a two-party call and connecting the two ends.

SIPAuthServe is open-source software that provides SIP authentication services. It processes the SIP REGISTER requests, which are generated by the OpenBTS every time an MS tries to join the mobile network. SQLite is a service that performs database functions in a SDR network [15], and allows network logs/files to be opened using a browser [20]. Figure 3 shows the block diagram of how the client accesses the database functions in SQLite. SMQueue is the store-and-forward message control unit that manages and handles message and messaging queues in the network. It uses logs and relevant functions to control the message flow and traffic. Table II shows a sample SMQueue entry table.

OpenBTS converts all signalling and media data being transferred among the various software dependencies to Session Initiation Protocol (SIP) and Real Time Protocol (RTP) messages as shown in Figure 4. Figure 5 shows the Command Line Interface (CLI) of the OpenBTS on Ubuntu OS. It has a series of commands that are used to configure and monitor the status of the network. For instance, the tmsis command generates a table that contains the list of connected MSs with their International Mobile Subscriber Identity (IMSI), International Mobile Equipment Identity (IMEI) and other parameters as shown in Figure 6.

D. NomadicBTS WebApp

The NomadicBTS WebApp is a major developmental contribution of this work to the SDR ecosystem. The WebApp-V1 architecture is composed of two core blocks: the frontend for user interactions and the backend for core functionality. Additionally, essential insights into the system are conveyed through three key diagrams: Figure 7 showcasing use cases, Figure 8 depicting class relationships, and Figure 9 outlining sequence interactions. This comprehensive combination provides a clear understanding of the NomadicBTS WebApp's design and functionality. The

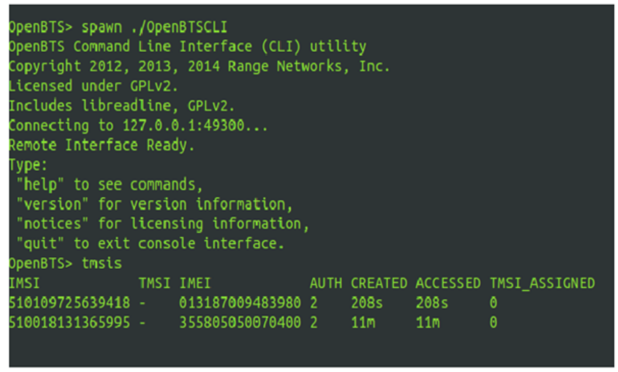


Figure 5. Sample of OpenBTS CLI on Ubuntu Operating System

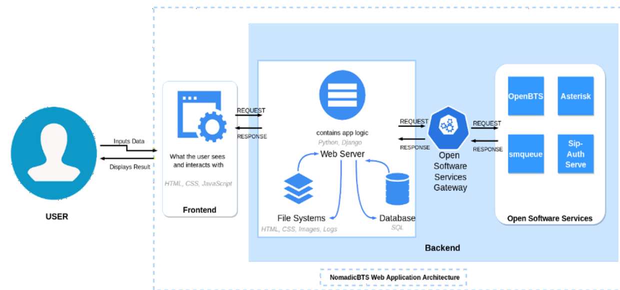


Figure 6. NomadicBTS WebApp Architecture

application employs a request-response model to transmit administrators' user requests for configuring, monitoring, and maintaining the network. Specifically, it allows an administrator to interface between the USRP and the open software services without the command line.

E. NomadicBTS Billing Service

This section focuses on creating a billing service for controlling, monitoring, recording, and charging calls and

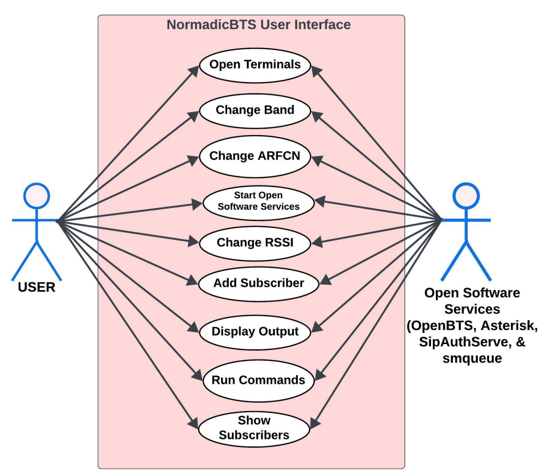


Figure 7. NomadicBTS WebApp Use Case Diagram

TABLE II. Sample Table in the SMQueue Database

Queue Type	Queue	Queue Type	Queue
11	Dist1_PortVeh_Queue	31	Dist3_PortVeh_Queue
12	Dist1_ContToOther_Queue	32	Dist3_ContToOther_Queue
13	Dist1_RoadVeh_Queue	33	Dist3_RoadVeh_Queue
14	Dist1_ContToRoad_Queue	34	Dist3_ContToRoad_Queue
21	Dist2_PortVeh_Queue	71	Port_PortVeh_Queue
22	Dist2_ContToOther_Queue	72	Port_ContToOther_Queue
23	Dist2_RoadVeh_Queue	73	Port_RoadVeh_Queue
24	Dist2_ContToRoad_Queue	74	Port_ContToRoad_Queue

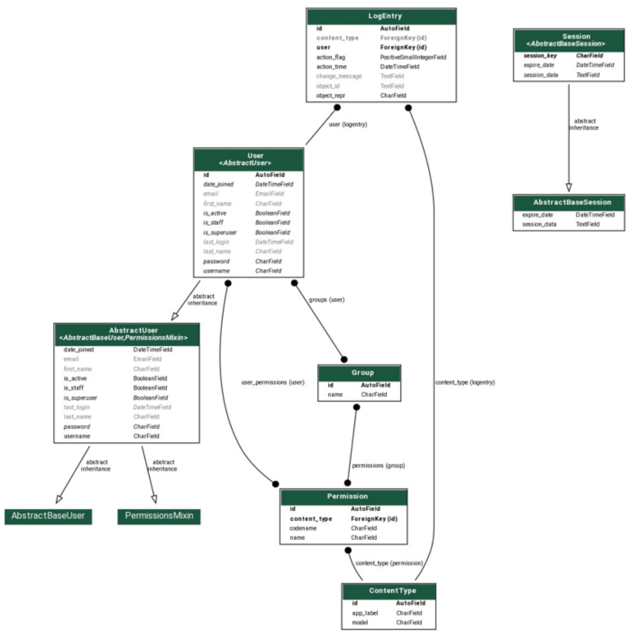


Figure 8. NomadicBTS WebApp Class Diagram

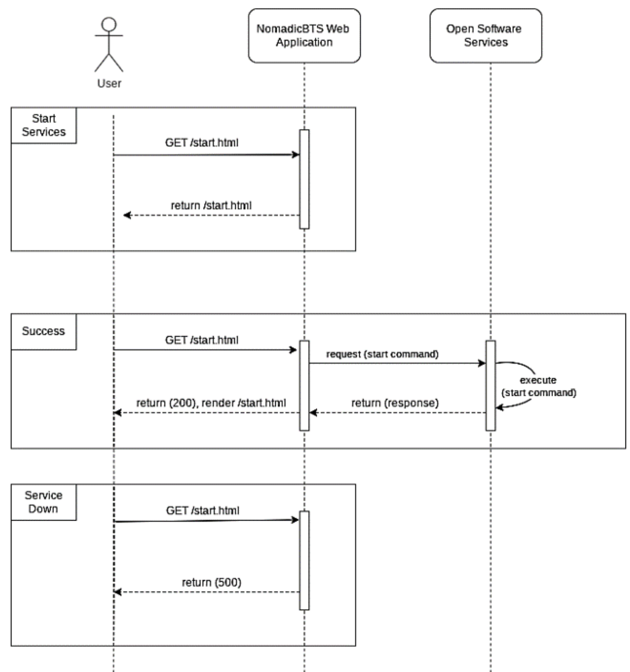


Figure 9. NomadicBTS WebApp Sequence Diagram

texts over the NomadicBTS network. It involves defining requirements, designing components, integration with voice/SMS services, and performance testing. The billing system includes charging for network-in-a-box calls and, for call monitoring, the ability to terminate calls without credit. Non-functional aspects encompass security, usability, interoperability, scalability, and data storage [34]. Figure 10 illustrates the OpenBTS system's connection network, while Figure 11 depicts the billing service's architecture, which includes the various modules and their interconnection. The detailed description of the various modules is presented in [35]. The system model is depicted with graphical representations which are based on the Unified Modelling Language (UML). The system operation is shown using a sequence diagram, which shows the interactions between items chronologically as presented in Figure 12.

The billing system monitors asterisk channels for active connections. When a subscriber initiates a mobile call, an asterisk channel activates. The billing system identifies the

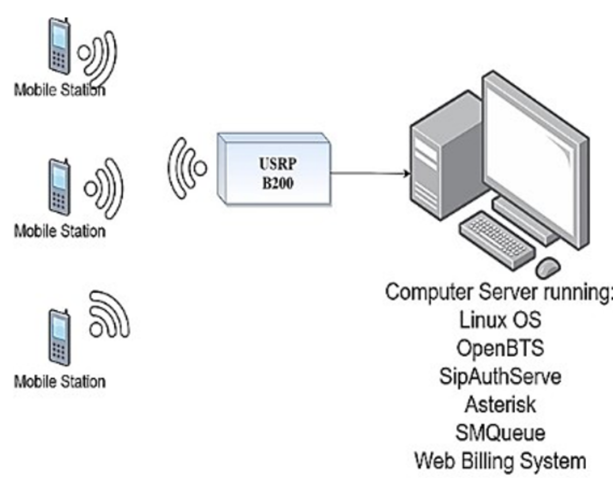


Figure 10. Connection Scheme of the OpenBTS Network

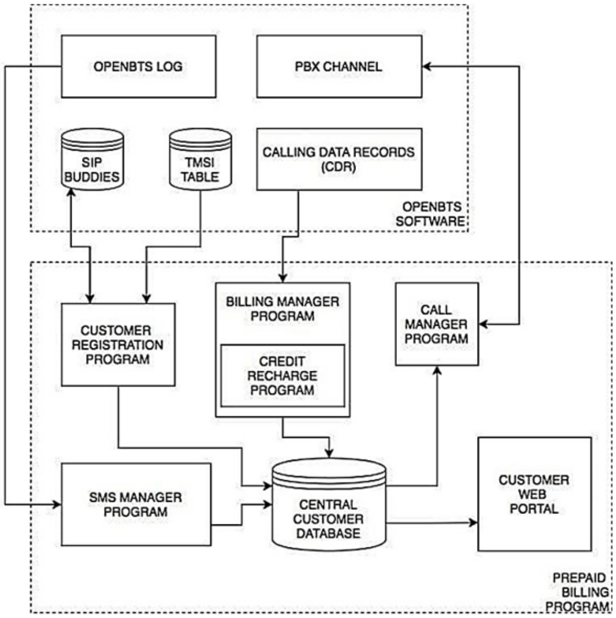


Figure 11. The Architecture of the Prepaid Billing Service

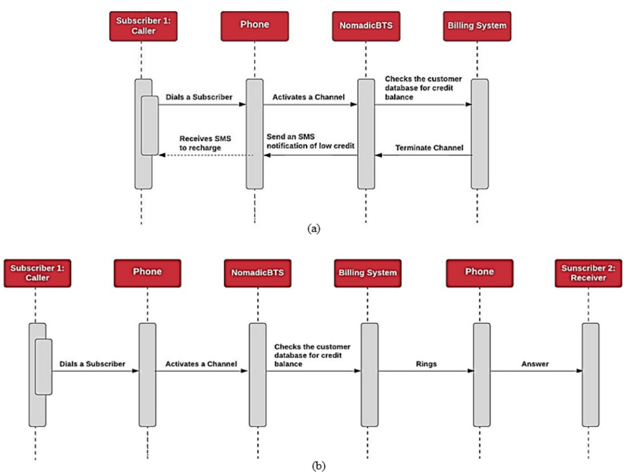


Figure 12. Sequence Diagram (a) With Zero Credit Balance (b) With Non-Zero Credit Balance

channel, retrieves call details, and queries the customer database for the caller's balance. If the balance is zero, the system ends the call, sends an SMS, and if not, allows communication. For non-zero balances, the system calculates call duration based on the 12 kobo per second rate. The flowchart in Figure 13 explains in detail the actions being taken by the billing system and the conditions it takes to get the desired goal.

F. Experimental Setup

The laboratory experimental setup of the NomadicBTS-2 platform is shown in Figure 14, while Table III presents the system hardware/software requirements. The Figure shows an HP-ProBook-440-GS laptop running an Ubuntu

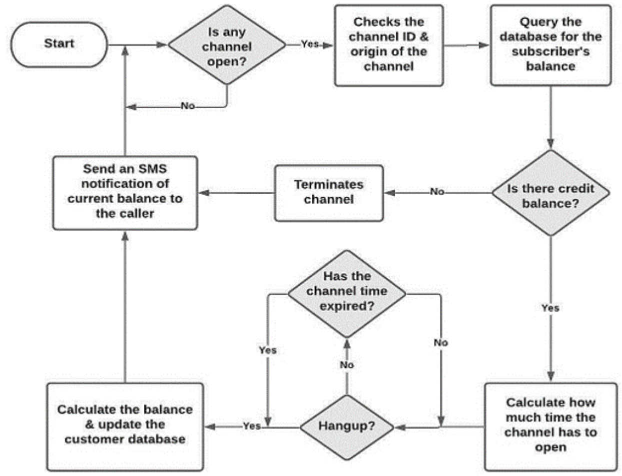


Figure 13. Flowchart of the Billing Service

Figure 14. Experimental Setup of NomadicBTS

16.04 Linux OS with an Intel Core i5-321OM, 8 GB Random Access Memory (RAM) and a processor speed of 2.50 GHz. The HP Laptop runs all the software backend services as well as the NomadicBTS WebApp. The USRP B200 with the VERT900 antenna is also seen connected to the Laptop via a USB 3.0 cable. This setup allows the MSs to perform over-the-air communication with the NomadicBTS-2 platform.

The USRP Hardware Driver (UHD), OpenBTS and other software services (i.e. Asterisk, SIPuthServe and SMQueue) can be configured and monitored graphically with the NomadicBTS WebApp. The graphical interface provides a better user experience than the Command Line Interface (CLI) configuration approach. For instance, Figure 15 shows the configuration of OpenBTS for GSM 1800 MHz using the CLI. The huge list of commands and having to type them every time and read through several lines of text is obviously cumbersome for network administrators.

3. Results

This section presents the performance testing results of the NomadicBTS-2, which include the link quality, accessibility (two-party call), SMS, and billing tests.

TABLE III. System Hardware and Software Requirements and Specifications

Hardware Requirements		Software requirements	
Components	Specifications	Requirements	Software
Computer server	SMQueue, SIPAuthServe	Operating system	Linux operating system
USRP	B200	Programming language	Python
Antenna	VERT900	Development framework	Django
Mobile phones	Android	GSM network	OpenBTS
SIM cards	Blank SIM cards		

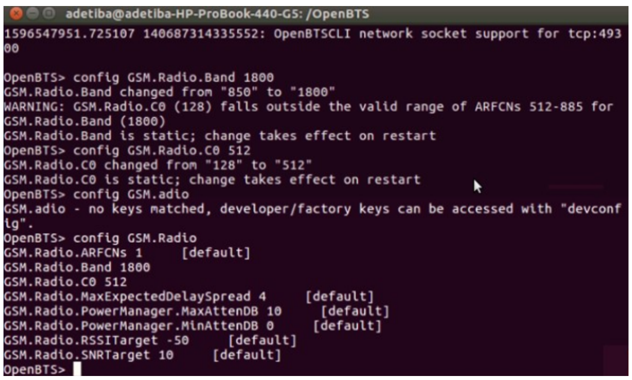
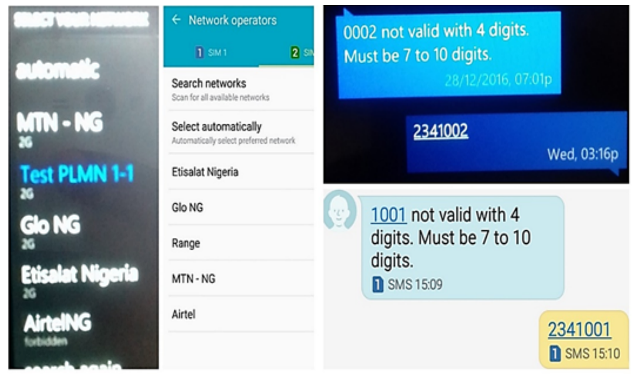


Figure 15. OpenBTS CLI-based Configuration



(a) (b)

Figure 16. (a) Screenshot of Network Search on the MSs, (b) Screenshot of MSISDN Assignment to Users

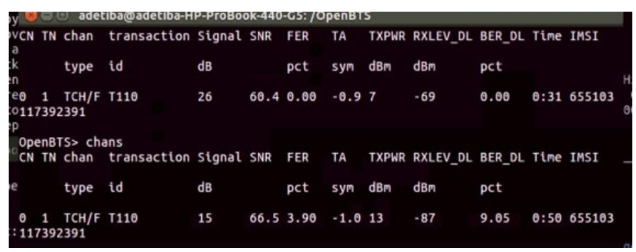


Figure 17. Channel Characteristics at a Time Instance

A. *Mobile Network Search*

The NomadicBTS-2 experimental cellular platform demonstrates the lack of complexity in starting the network to begin service provisioning for mobile users. Two MSs were powered on, and a network search was carried out on the two MSs. The network initialization revealed that the two MSs could find and connect on the Broadcast Control Channel (BCCH) transmitted by the experimental cell as shown in Figure 16 (a). The Test PLMN 1-1 is the ID broadcast by the NomadicBTS-2 cell. Furthermore, an identifier (MSISDN) was assigned to each of the MS by sending an SMS containing the desired number, which must be between 7-10 digits to a switching number - “101’, as shown in Figure 16 (b). The number assigned to the Samsung MS is “2341001,” and the number assigned to the Nokia MS is “2341002”.

B. *Measurement of Link Quality*

The ubiquitous use of mobile devices in the terrestrial telecommunications network has been a major motivation for mobile network operators to always perform up-to-date network monitoring and optimization to improve the quality of service (QoS) provisioning for subscribers. However, many factors still militate against the overall performance of the network vis-à-vis poor coverage, interference, noise, low throughput, and link failure, amongst others. The NomadicBTS is not immune to these radio link problems because the medium of signal propagation is unguided, hence, power dissipates over a distance.

To analyze the behavior of the network, the link quality test was performed on each of the frequency bands. It is noteworthy that the quality of a radio link is best evaluated when the traffic channel is in use at a time instance on the network as shown in Figure 17. A dial tone call was initiated, and the chains command was executed to measure the link quality as the MS moved away from the NomadicBTS. For each time instance, an active channel was observed; the uplink signal-to-noise ratio (SNR) and received signal level (RXLEVEL) decreased with distance (fading), while the transmit power (TXPWR) of the NomadicBTS increased. The results show that the higher the SNR, the better the link quality, and the lower the SNR, the weaker the link quality.

Similar to the traditional complex GSM system, the link quality determines the quality of experience perceived by mobile users – call success rate and latency. Table IV shows the link quality of GSM 1800 MHz while Table V presents

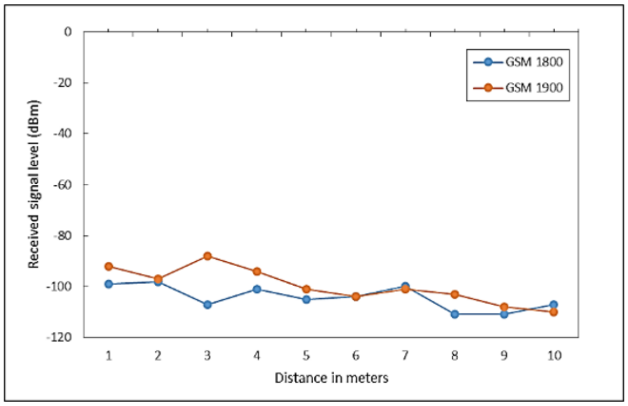


Figure 18. Received Signal Strength Measurement for the NomadicBTS

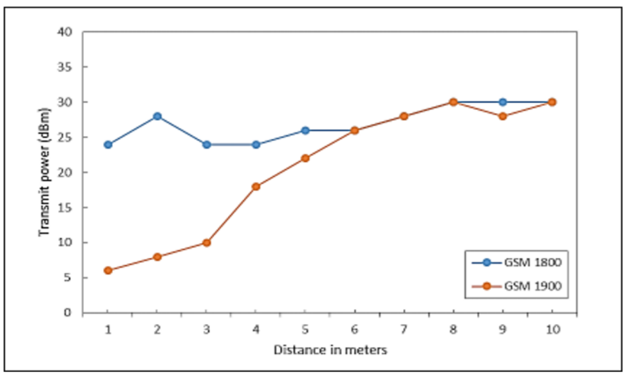


Figure 19. Transmit Power Measurement for the NomadicBTS

quality for GSM 1800 MHz bands as the mobile station traversed the coverage range of the NomadicBTS.

The link quality test reveals the performance of the two bands: GSM 1800 MHz and 1900 MHz. First, the received signal power attenuates over a distance; this is caused by the signal variation in the test environment (indoor) as the transmitted signals suffer from reflection and scattering before reaching the destination. This behavior is similar in both bands but more severe in the GSM 1800 MHz band, as shown in Figure 18. The relatively low RXLEVEL performance obtained stems from the fact that the transmit power of the NomadicBTS is designed for short-range communication, as observed in Figure 19 where the TXPWR is increased over a relatively small coverage (10 meters). Also, the GSM 1800 MHz does not provide an energy-efficient transmission as high TXPWR is required for the entire communication, in contrast to GSM 1900 MHz, where a low TXPWR is leveraged to give permissible voice transmission up to 5 meters of coverage.

This is a comparative justification of the relationship between RXLEVEL and TXPWR in mobile communications as shown in Figure 18 and Figure 19 respectively. Another link quality metric is the SNR, the ratio of the

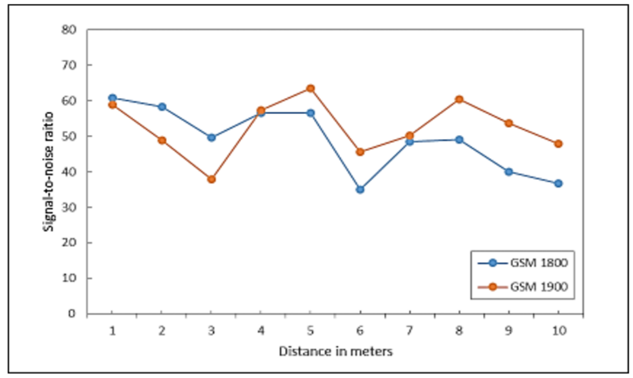


Figure 20. SNR Measurement for the NomadicBTS

signal power to the noise power. In mobile communications, the goal is to maximise the SNR for reliable data or voice transmissions [36]. The measurement of the SNR over a dedicated traffic channel reveals that the maximum signal power is stochastically distributed over distance as shown in Figure 20. The variation in the SNR is a function of the inherent channel noise caused by erroneous transmissions from other devices. However, since our approach is tailored towards exploiting the open-source technologies and SDR paradigm for short-range communication; in this study, the GSM 1900 MHz provides a more pragmatic solution leveraging the NomadicBTS.

C. Operation and Maintenance using NomadicGUI

As discussed in Section II, the web server responds to the user's requests, and loads objects such as scripts, images, and logs from the machine's file system.

The frontend of the NomadicGUI application presents to the user a modern interface rendered by a web browser. The web interface is achieved by a combination of HTML, CSS and Javascript. These scripts are delivered to the web browser in response to requests made by the user as shown in Figure 21. The operation and maintenance of the NomadicBTS is carried out efficiently using the NomadicGUI application; the administrator or support staff in a real-practice case is eased of writing multiple lines of network-level codes in the Ubuntu OS CLI.

The signal strength of the base station was monitored using the Network Signal Info Pro tool installed on the test phones. The signal strength was observed for the two frequency bands - GSM 1800 MHz and 1900 MHz. The received signal strength at a time instance when operating on the GSM 1800 MHz and 1900 MHz, as shown in Figure 22 (a) and (b), are -85 dBm and -63 dBm respectively. For SMS routing and delivery, the SMSQueue was started, and SMS tests were performed. To confirm that the SMQueue works, a "hello" message was sent from the network to MS1 from the defined network switching number "120". The next test performed was the echo SMS. To perform an echo SMS, a message "NETWORK TESTING" was sent from the MS1

TABLE IV. Link Quality Summary for NomadicBTS for GSM 1800 MHz

Distance (m)	Rxlevel (dBm)	TxPower (dBm)	SNR
1	-99	24	60.9
2	-98	28	53.8
3	-107	24	49.7
4	-101	24	56.6
5	-105	26	56.5
6	-104	26	35.1
7	-100	28	48.5
8	-111	30	49
9	-111	30	40

TABLE V. Link Quality Summary for NomadicBTS for GSM 1900 MHz

Distance (m)	Rxlevel (dBm)	TxPower (dBm)	SNR
1	-92	6	59
2	-97	8	48.8
3	-88	10	37.9
4	-94	18	57.4
5	-101	22	63.6
6	-104	26	45.7
7	-104	28	50.2
8	-103	30	60.5
9	-108	28	53.2

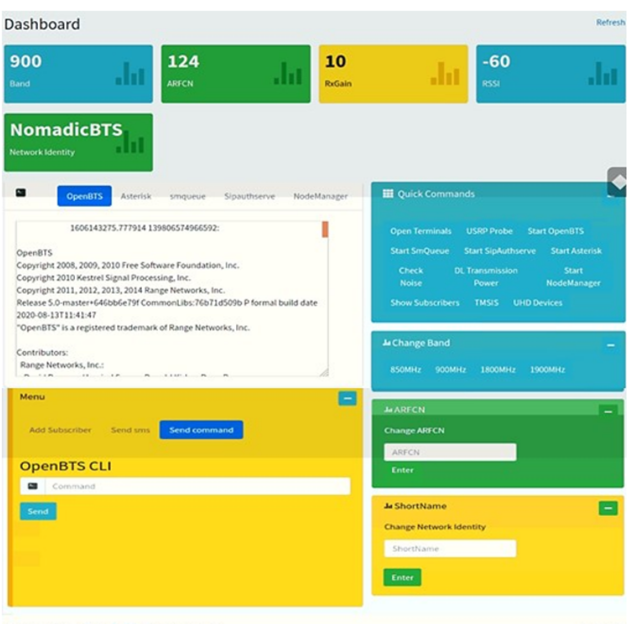


Figure 21. NomadicGUI Frontend

to a network switching number 411. The switching numbers are randomly generated for each session, and stored in the SMQueue, which echoes back whatever message it receives. The SMS test and echo SMS are shown in Figure 23 (a) and (b) respectively.

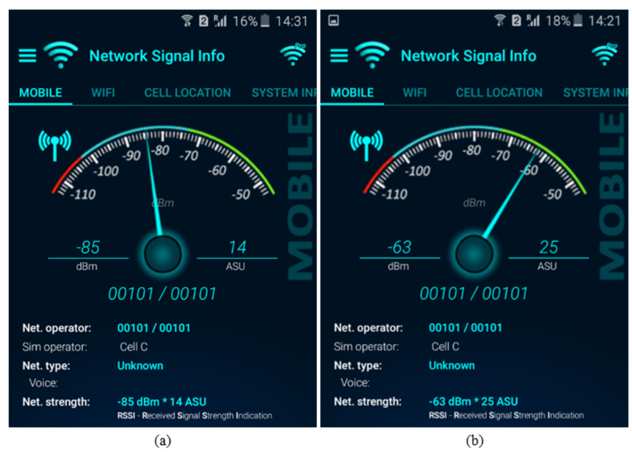
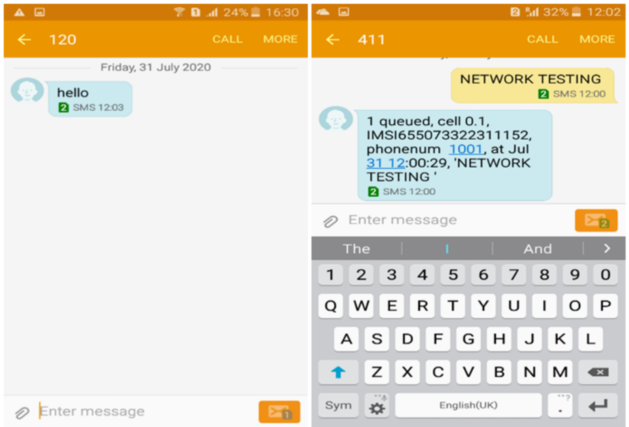


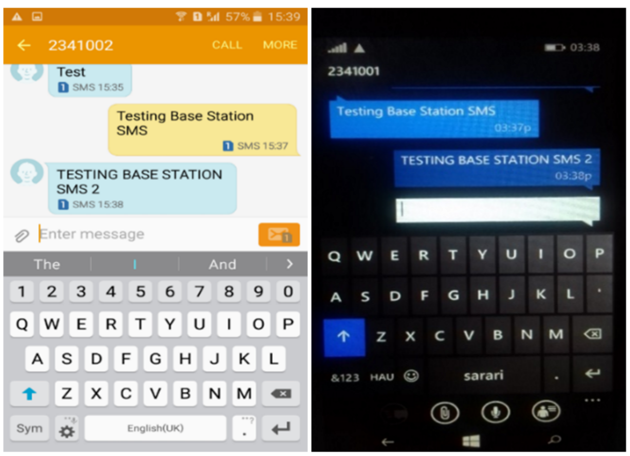
Figure 22. (a) RSS on GSM 1800 MHz, (b) RSS on GSM 1900 MHz

The MS-to-MS (two-party) communication flow shows good-level performance for the SMS tests, and call tests. This was carried out after assigning unique MSISDN to each mobile subscriber. For the two-party SMS tests, a message "Testing Base Station SMS" was sent from the MS1 to MS2, while a separate message "TESTING BASE STATION SMS 2" was sent from the MS2 to the MS1 as shown in Figure 24 (a) and (b), and for the call tests the two MSs exchanged communication over a period of 18 seconds, as shown in Figure 25.



(a) (b)

Figure 23. (a) SMS Test, and (b) Echo SMS Test



(a) (b)

Figure 24. Two-party SMS Test

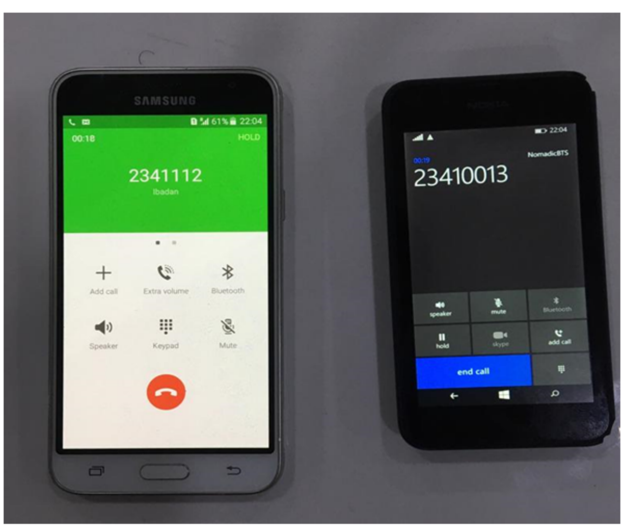
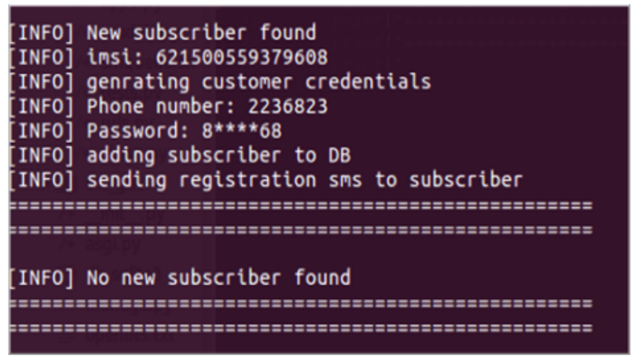


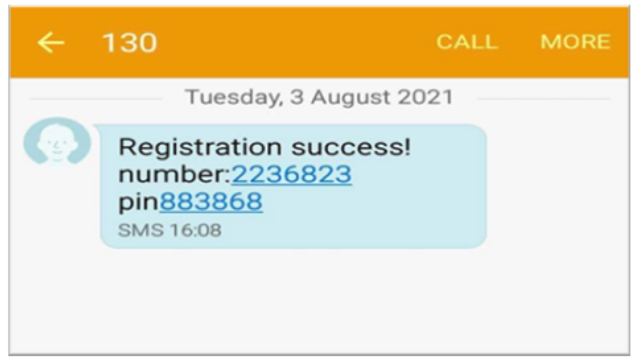
Figure 25. Two-party Call Test



```
[INFO] New subscriber found
[INFO] imsi: 621500559379608
[INFO] genrating customer credentials
[INFO] Phone number: 2236823
[INFO] Password: 8****68
[INFO] adding subscriber to DB
[INFO] sending registration sms to subscriber
================================================
================================================

[INFO] No new subscriber found
================================================
================================================
```

(a)



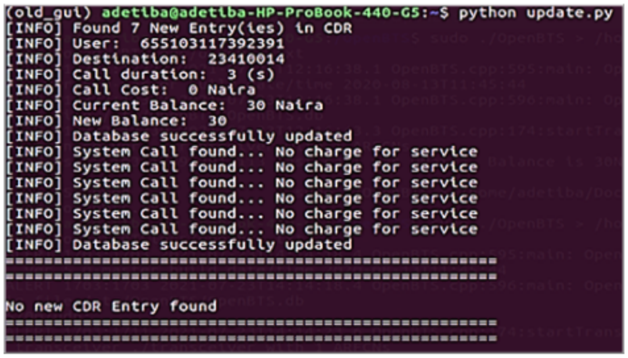
(b)

Figure 26. Screenshot of (a) the Output of the Registration Script (b) SMS Notification for a Successful Registration

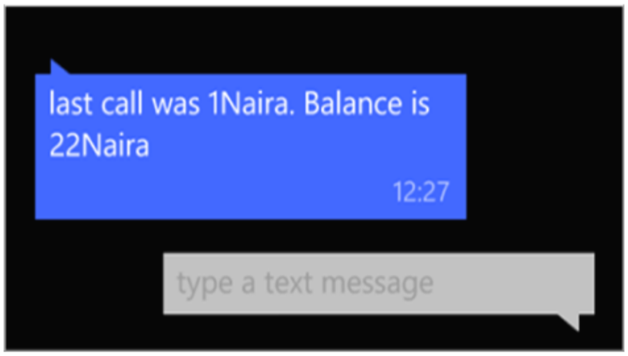
D. Billing System Performance

During customer registration, the Temporary Mobile Subscriber Identity (TMSI) database logs the International Mobile Subscriber Identity (IMSI) number of new users. The program adds this IMSI to the sipbuddies database, assigns a number to the subscriber, and generates a password for web app access. The program executed successfully, populating the sipbuddies table and generating credentials. Figure 26 (a) and (b) display the process. Moreover, it acquires call details like origin, destination, and caller's balance. Channel monitoring is continuous, assessing active channels and IDs. This program verifies active channels' call origin and destination. It calculates call costs, deducts from the client's balance, and sends an SMS with a new balance post-call. It ends calls on balance depletion. The call duration and the 12 kobo/sec rate notification are through SMS. Figure 27 (a) and (b) exhibit billing script output and SMS balance update.

This software manages SMS billing and fee deductions from subscriber balance. It successfully charged SMS senders based on call duration and a 1 Naira/SMS fee, then sent an SMS balance update. Figure 28 (a) and (b) showcase the SMS billing script output and balance update. The software also generates recharge numbers with various costs, deleting used vouchers.

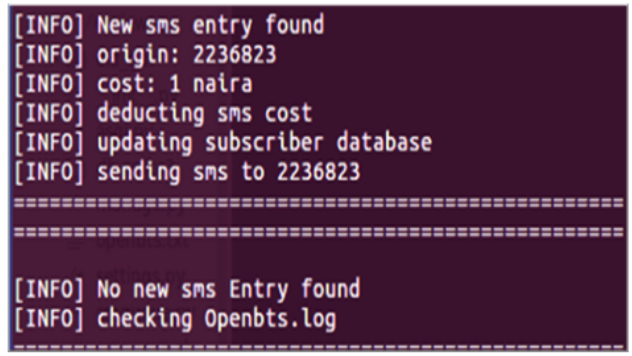


(a)

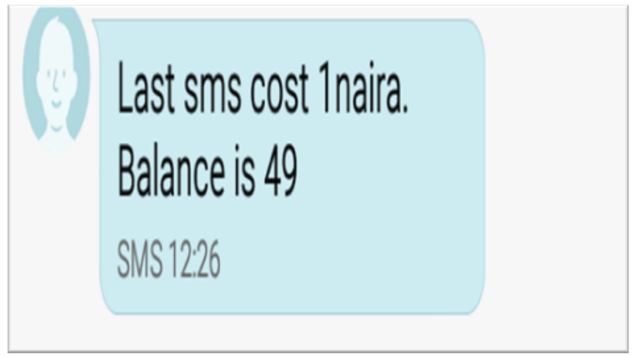


(b)

Figure 27. Screenshot of (a) the Billing System (b) SMS Showing the Bill of a Call



(a)



(b)

Figure 28. Screenshot of (a) the Output of the SMS Billing Script (b) SMS Notification of New Balance Sent to Subscriber

4. Conclusion

This study has presented a comprehensive overview of the NomadicBTS system, encompassing its design, operational functionality, billing, and management through the utilization of the NomadicGUI (v1.0). While the initial challenge of ensuring seamless integration of various open-source software components due to interoperability issues was significant, it was successfully overcome. The findings underscore the system's potential to effectively address the challenges inherent in mobile communication networks operating in rural and remote locations. While GSM remains the dominant wireless communication technology, issues like network congestion, base station damage, and base station unavailability have hindered the accessibility of fundamental telecommunication services. In this context, our innovative approach emerges as a solution that is not only cost-effective but also attuned to mobility requirements, offering a marked departure from traditional GSM architecture.

The significance of this approach lies in its adaptability and potential for widespread application. Telecommunication companies can leverage the insights provided in this research to deploy a scalable mobile Software-Defined Radio Base Transceiver Station (SDR-BTS) version. This streamlined and intuitive design facilitates the implementation and holds the potential to revolutionise access to telecommunication services, particularly in areas where conventional infrastructure falls short. Additionally, governmental agencies and other stakeholders can readily adopt and expand upon this system, particularly in scenarios demanding swift and efficient communication solutions, such as ad-hoc or mission-critical situations. The demonstrated feasibility of this approach establishes a promising avenue for enhancing connectivity in underserved regions and contributing to the advancement of communication networks in challenging contexts.

References

[1] J. Andrews, S. Buzzi, W. Choi, S. Hanly, A. Lozano, A. Soong, and J. Zhang, "What will 5G Be?" *IEEE Journal on selected areas in communications*, vol. 32, pp. 1065–1082, 2014.

[2] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five Disruptive Technology Directions for 5G," *IEEE communications magazine*, vol. 52, pp. 74–84, 2014.

[3] J. Pedreno-Manresa, P. Khodashenas, M. Siddiqui, and P. Pavon-Marino, "On The Need of Joint Bandwidth and NFV Resource Orchestration: A Realistic 5G Access Network Use Case," *IEEE Communications Letters*, vol. 22, pp. 145–148, 2017.

[4] M. Sadiku and C. Akujuobi, "Software-defined radio: a brief overview," *IEEE Potentials*, vol. 23, pp. 14–15, 2004.

[5] E. Thompson, N. Clem, I. Renninger, and T. Loos, "Software-

Defined GPS Receiver on USRP-platform," *Journal of Network and Computer Applications*, vol. 35, pp. 1352–1360, 2012.

[6] T. Tozer and D. Grace, "High-altitude platforms for wireless communications," *Electronics & Communication Engineering Journal*, vol. 13, pp. 127–137, 2001.

[7] D. Kune, H. N. Koelndorfer, J., and Y. Kim, "Location Leaks on the GSM Air Interface," *ISOC NDSS*, pp. 205 190–205 205, 2012.

[8] I. Haq, Z. Rahman, S. Ali, and E. Faisal, "GSM Technology: Architecture, Security and Future Challenges," *International Journal of Science Engineering and Advance Technology*, vol. 5, pp. 70–74, 2017.

[9] J. Mitola, "The Software Radio Architecture," *IEEE Communications magazine*, vol. 5, pp. 26–38, 1995.

[10] Mitola, J.I., "Cognitive Radio. An Integrated Agent Architecture for Software Defined Radio," Ph.D. dissertation, KTH - Royal Institute of Technology, Stockholm, 2000.

[11] E. Adetiba, V. Matthews, S. John, S. Popoola, and A. Abayomi, "NomadicBTS: Evolving Cellular Communication Networks with Software-defined Radio Architecture and Open-source Technologies," *Cogent Engineering*, vol. 5, pp. 1–15, 2018.

[12] E. Adetiba, F. Olaloye, A. Abayomi, N. Faruk, S. Moyo, O. Obiyemi, and S. Thakur, "Compact Automatic Modulation Recognition using Over-the-air Signals and FOS Features," *Bulletin of Electrical Engineering and Informatics*, vol. 11, pp. 2013–2024, 2022.

[13] Z. Tong, M. Arifianto, and C. Liau, "Wireless Transmission using Universal Software Radio Peripheral," *In 2009 International Conference on Space Science and Communication*, pp. 1–5, 2009.

[14] M. Wibisono, S. Priatna, T. Juhana, and N. Rachmana, "On the design and development of flying bts system using balloon for remote area communication," *In 2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA)*, pp. 1–5, 2017.

[15] Hafidudin, M. Rizal, and D. Ramadan, "Implementation of USRP (Universal Software Peripheral Radio) as OpenBTS for Quadruple Play Services," *In Journal of Physics: Conference Series*, vol. 1364, p. 012037, 2019.

[16] R. Helbet, V. Monda, A. Bechet, and P. Bechet, "Low Cost System for Terrestrial Trunked Radio Signals Monitoring Based on Software Defined Radio Technology and Raspberry Pi4," *2020 International Conference and Exposition on Electrical And Power Engineering (EPE)*, pp. 438–441, 2020.

[17] F. Chiper, A. Martian, C. Vladeanu, I. Marghescu, R. Craciunescu, and O. Fratu, "Drone detection and defense systems: Survey and a Software-Defined Radio-Based solution," *Sensors*, vol. 22, p. 1453, 2022.

[18] A. Martian, F. Chiper, R. Craciunescu, C. Vladeanu, O. Fratu, and I. Marghescu, "Rf based UAV detection and defense systems: Survey and a novel solution," *In Proceedings of the 2021 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, pp. 1–4, 2021.

[19] D. Molla, H. Badis, L. George, and M. Berbineau, "Software defined radio platforms for wireless technologies," *IEEE Access*, pp. 26 203–26 229, 2022.

[20] J. Muslimin, A. Asnawi, A. Ismail, and A. Jusoh, "SDR-based transceiver of Digital Communication System using USRP and GNU Radio," *In 2016 International Conference on Computer and Communication Engineering (ICCCE)*, pp. 449–453, 2016.

[21] N. Prasannan, G. Xavier, A. Manikkoth, R. Gandhiraj, R. Peter, and K. Soman, "OpenBTS based microtelecom model: A socio-economic boon to rural communities," *In 2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s)*, pp. 856–861, 2013.

[22] K. Aggrawal and K. Vachhani, "Reconfigurable Cellular GSM Network using USRP B200 and OpenBTS for Disaster-hit Regions," *In 2017 IEEE 13th Malaysia International Conference on Communications (MICC)*, pp. 141–146, 2017.

[23] Y. Nugroho, R. Sari, and R. Harwahyu, "Performance Comparison of GPRS and LTE Telecommunication Network using OpenAir-Interface and OpenBTS with USRP," *In 2020 2nd International Conference on Industrial Electrical and Electronics (ICIEE)*, pp. 197–204, 2020.

[24] J. Uribe and E. Guillen, "Vulnerability Assessment for IoT Nodes Using OpenBTS and Software Defined Radios," *In 2020 3rd International Conference on Signal Processing and Information Security (ICSPIS)*, pp. 1–4, 2020.

[25] F. Partiansyah, S. Kusmaryanto, R. Ambarwati, and S. Pramono, "Experimental Study of USRP N210 as Simple GSM OpenBTS 5.0 for Remote Areas," *In 11th Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS)*, pp. 185–190, 2022.

[26] T. Putri and T. Juhana, "Mobile-openBTS implementation of natural disaster victims search," *In 3rd International Conference on Wireless and Telematics (ICWT)*, pp. 149–154, 2017.

[27] L. Behan, L. Orcik, F. Rezac, I. Baronak, and J. Lin, "Prepaid voice services based on OpenBTS platform," *In proceedings of the 3rd Czech-China Scientific Conference*, 2017.

[28] O. Castro, "Network GPRS prototype based on SDR and OpenBTS, as an IoT-lab Testbed," *In seventh international conference on software defined systems (SDS)*, pp. 14–19, 2020.

[29] E. Bratu and B. Harper, "Open Source Mobile Network Using OpenBTS and USRP 9," *Venture: The University of Mississippi Undergraduate Research Journal*, vol. 2, p. 12, 2017.

[30] M. Quispe, F. Saccaco, J. Olivares, J. Samaniego, and S. Inca, "On the Use of Open Air Interface and OpenBTS for Mobile Communications Education in Telecommunications Laboratories," *In IEEE XXX International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, pp. 1–6, 2023.

[31] Ettus, "Ettus research b200mini sdr platform," *Available online: https://www.ettus.com/products/*, (accessed on 25 January 2022).

[32] C. Alenoghena and J. Emagbetere, "Base station placement challenges in cellular networks: The Nigerian experience," *In 2012 IEEE 4th International Conference on Adaptive Science Technology (ICAST)*, pp. 7–11, 2012.

[33] A. Apvrille, "An OpenBTS GSM replication jail for mobile mal-

ware," *In Virus Bulletin Conference, VB2011*, pp. 86–94, 2012.

[34] M. Saitwal, M. Khan, and U. Desai, "Cognitive GSM OpenBTS," *in IEEE 11th International Conference on Mobile Ad Hoc and Sensor Systems*, pp. 529–530, 2014.

[35] P. Uzoatuegwu, "Development of a billing web service for the. NomadicBTS Cellular Network-in-a-box," *Covenant University*, 2021.

[36] S. Friedland and C. Tan, "Maximizing Sum Rates in Gaussian Interference-limited Channels," *Optimization and Control*, pp. 1–24, 2008.