Tuning into the Web: A Low-Cost Access Solution for Developing Countries

Ayush Pandey
New York University Abu Dhabi
Abu Dhabi, UAE
ayush.pandey@nyu.edu

Rohail Asim
New York University Abu Dhabi
Abu Dhabi, UAE
rohail.asim@nyu.edu

Jean Louis K. E. Fendji
University of Ngaoundere
Ngaoundere, Cameroon
lfendji@univ-ndere.cm

Talal Rahwan
New York University Abu Dhabi
Abu Dhabi, UAE
talal.rahwan@nyu.edu

Matteo Varvello
Nokia Bell Labs
Holmdel, NJ, USA
matteo.varvello@nokia.com

Yasir Zaki
New York University Abu Dhabi
Abu Dhabi, UAE
yasir.zaki@nyu.edu

Abstract

Over a quarter of the global population remains without access to the Internet in 2026, primarily due to cost and limited infrastructure in developing regions. This gap is becoming more consequential as access to information increasingly depends on AI systems that presume reliable, high-bandwidth connectivity. In response, we present WebFM, a low-cost, scalable Web and AI access solution that builds on widely deployed infrastructure in developing regions: FM radio for downlink broadcasting and SMS for personalized uplink. Prior work on web-over-FM is limited to simulations, short-range experiments, and lossless links, leaving the effects of geographically induced signal variability largely unexplored. WebFM addresses this gap through several system-level innovations to effectively transmit Web content and enable LLM interactions over sound over FM radio, in a reliable and compressed form. For example, we introduce a loss-resilient file format for encoding webpages and modify Android to leverage built-in FM tuners, allowing deployment across a wide range of devices without requiring root access. We enable ChatGPT interactions by allowing users to submit prompts over SMS and receive model responses via FM broadcast. Finally, we preemptively transmit popular pages and a subset of their internal pages while the server is idle, improving user experience during peak hours. We deployed WebFM at an FM radio station in Cameroon for six weeks with 30 participants. Our evaluation shows a sustained downlink throughput of 10 kbps, less than 20% loss for most transmissions with signal strength above -90 dBm, and strong user engagement across both web browsing and ChatGPT interactions.

CCS Concepts

• **Networks** → **Network design principles**; • **Information systems** → *World Wide Web*.

Keywords

FM radio, World Wide Web, Developing regions, Digital divide

ACM Reference Format:
Ayush Pandey, Rohail Asim, Jean Louis K. E. Fendji, Talal Rahwan, Matteo Varvello, and Yasir Zaki. 2026. Tuning into the Web: A Low-Cost Access Solution for Developing Countries. In *ACM SIGCOMM 2026 Conference (SIGCOMM '26), August 17–21, 2026, Denver, CO, USA.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3789240.3829209

SIGCOMM '26, Denver, CO, USA

ACM ISBN 979-8-4007-2467-1/26/08
https://doi.org/10.1145/3789240.3829209

1 Introduction

The internet has fundamentally reshaped societies worldwide, driving economic growth, fostering new industries, and becoming indispensable in education, healthcare, and employment. Yet, despite its profound influence, 26.7%—over 2.21 billion people—remain disconnected in 2026 [81]. This *digital divide* is not only due to a lack of infrastructure. According to GSMA Intelligence [47], 3.1 billion people live within the coverage area of a mobile broadband network but remain disconnected because they cannot afford the necessary devices and data plans to get online.

The offline population is especially concentrated in developing regions. For example, India has 440 million people (30%) offline, Pakistan 139 million (54.4%), and Nigeria 130 million (54.5%) [81]. In the Central African Republic and South Sudan, more than 85% of the population remains without internet [81]. The consequences of being "unconnected" go beyond missed economic opportunities; it severely limits access to essential services such as education, healthcare, and financial resources, further increasing existing inequalities.

Several initiatives have attempted to bridge this gap. Starlink [13] provides global coverage through low earth orbit (LEO) satellites. Google's Project Loon [6] sought to deliver internet via high-altitude balloons, while Facebook's Aquila [8] aimed to use solar-powered drones (both ultimately discontinued due to maintenance costs and scalability challenges). Project Taara [17] is a recent initiative that transmits data using laser beams over long distances (20 km) at high speeds (20 Gbps). Unfortunately, these efforts remain prohibitively expensive in most developing regions. For example, in Zambia, classified as one of the UN's least developed countries [86], Starlink's $40 monthly subscription (plus $180 hardware fee) is beyond reach for many compared to the country's $108 monthly GNI per capita.

In contrast to these approaches, many developing countries already possess a widely deployed communication infrastructure: FM radio, offering long range, strong penetration even in topographically diverse regions [18, 20, 70]. FM reception is free for end users, and its broadcast nature allows reaching large number of users simultaneously. Motivated by these advantages, Pandey et al. [68] introduced an initial framework to transmit webpages over FM radio. While promising, the study is positioned as a proof-of-concept and evaluates controlled, short-range links through simulations and limited experiments. In particular, audio is delivered to FM-capable phones via an external radio using a lossless auxiliary cable link, without utilizing internal FM radio hardware on commodity phones or examining geographically induced signal variability that arises in broadcast reception over hundreds of meters due to user location, antenna placement, obstruction, and multipath propagation. It also does not address the systems questions required for sustained shared access, such as request ingestion, scheduling, caching, and long-term station operation. In this paper, we build WebFM. Our main contributions are:

System Architecture. We design an end-to-end server-client architecture of WebFM, which includes modules for content rendering, prefetching, encoding, transmission, decoding, and error correction, required to operate a shared, lossy access service in the real world. Unlike prior proof-of-concept systems, WebFM treats corrupted and missing frames as a central design constraint, introducing a loss-resilient file format (in Section 6) to recover data over highly constrained, lossy FM links.

FM Tuning on Android. We utilize the internal FM tuner found in many Android devices to receive and decode WebFM transmissions. We do this by modifying LineageOS, an open-source Android operating system, to enable programmatic access to the FM hardware, allowing other user-scoped apps to control FM chip tuning and access the received audio stream directly, without needing to root the device.

LLM Support. We use the same audio modulation pipeline to allow users to query models like ChatGPT via SMS and receive coherent responses without internet access. We show that LLM responses, due to their smaller size, can be transmitted significantly faster than full webpages.

Real-world Deployment. We report on a six-week deployment at a live FM radio station in Cameroon. During this period, 30 participants used the WebFM app to request webpages and interact with ChatGPT via SMS. Our evaluation shows that WebFM can sustain a transmission rate of 10 kbps, with less than 20% data loss for a majority of transmissions when RSSI is above -90 dbM. Finally, in Section 4.3, we analyze the economic viability of WebFM, using Cameroon as a representative case study.

2 Related Work

Web Simplification. Habib et al. [49] proposed a framework that dynamically adapts webpage complexity based on network conditions. Muzeel [55] removes unused JavaScript, while MAML [69] introduces a minimalist specification language omitting JavaScript and CSS. Klotski [29] prioritizes user-relevant content, Shandian [92] restructures loading via split-browser design, and Polaris [64] accelerates rendering using dependency graphs. Several other tools such as compression proxies [9, 44], reader modes [12, 43], JavaScript cleaners [31–33], and redundant code removers [54, 60] have also explored reducing webpage complexity to improve performance under limited connectivity. Kiwix [40] is a recent open-source project that creates highly compressed copies of entire websites and fits into a single file. When offline, users read these files from their device storage to browse content. Though effective under constrained bandwidth, these methods require (at least) basic internet access, which is unavailable or unaffordable in many rural areas.

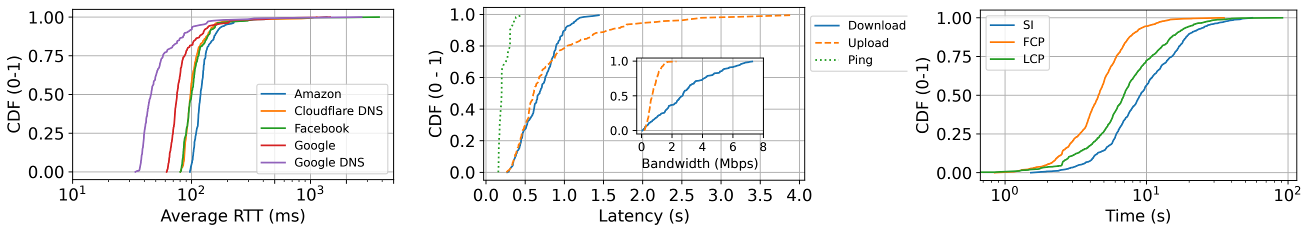
Data over FM. Radio Data System (RDS, or RBDS in the US) is the earliest FM subcarrier standard for embedding auxiliary digital data within FM signals. RevCast [79] leverages RDS for certificate revocation, invalidating TLS/SSL certificates before expiration. Rahmati et al. [72] developed RDS-Link, achieving 17.4 bytes/sec throughput for delay-tolerant applications in areas lacking cellular coverage. Vigil et al. [90] investigated using RDS to deliver social network content at 534 KB per hour to underserved communities. Bozomitu et al. [26, 27] broadcast driver warnings using DSSS-CDMA transmissions, where GPS coordinates serve as CDMA keys for decoding relevant imagery.

Microsoft's MSN Direct [10] (2003-2012) used the 67.65 kHz subcarrier leased from commercial broadcasters, delivering 12 kbps per tower (over 100 MB per day per city). This subscription service provided traffic and weather updates to Garmin GPS devices but suffered from cellular broadband competition and downlink-only communication limitations. μStation [94] demonstrated peer-to-peer broadcasting using mobile FM hardware on Nokia N900, achieving up to 20 kbps through pulse amplitude modulation for location-based profile sharing. However, mobile broadcast offers shorter transmission range and lacks support in recent smartphone chipsets.

Beyond FM subcarriers, several prior works [22, 34, 52, 57, 74, 78] and open-source projects [5, 7, 11] have investigated transmitting data through both audible and inaudible (ultrasound) frequencies, *i.e.,* above 18 kHz. Early studies [52, 57, 78] primarily target the near-ultrasound range (*i.e.,* 17–22 kHz) and report relatively low data rates over varying distances: 15 bps at distances up to 25 m [57], 16 bps at 2.7 m [52], and 2.76 kbps at 1 m [78]. AudioCast [34] is a recent study that demonstrates ultra-low-power FM transmission with audio-band Frequency-Shift Keying (FSK [80]), achieving 368 bps up to 130 m using commodity FM receivers. In contrast, BackDoor [74] achieves higher throughput (4 kbps at distances up to 1 m) by exploiting microphone hardware non-linearities to induce a "shadow" signal in the audible spectrum. Building on this concept, BatComm [22] incorporates Orthogonal Frequency Division Multiplexing (OFDM [58]) with an ultrasonic carrier, enabling data transmission across multiple orthogonal subchannels. As a result, BatComm achieves significantly higher rates: 47 kbps at 10 cm and 17 kbps at 2.3 m. Similarly, SONIC [68] uses OFDM to encode audio and transmit it with a Raspberry Pi acting as an FM transmitter for short-range experiments from 10 cm to 1.1 m. SONIC does not address FM radio hardware utilization on commodity phones, which limits deployability. It also does not evaluate a shared multi-user access service, leaving open how such a system would perform under scheduling contention. Appendix C contrasts the full server–client architecture of WebFM with SONIC. We further compare existing data-over-FM methods in Table 1 and demonstrate how WebFM outperforms prior approaches.

3 Motivation

Affordability and QoE in Rural Areas. Mobile internet adoption is rapidly increasing in low- and middle-income countries [47], yet ensuring affordability and a good quality of experience (QoE) remains a challenge. In many remote areas, even when mobile broadband is available, data costs are prohibitively high relative to



(a) RTT across cloud service providers.

(b) Speedtest metrics (latency and bandwidth).

(c) Web performance metrics (SI, FCP, LCP).

Figure 1: Network performance of MTN, the best available mobile network provider in Cameroon.

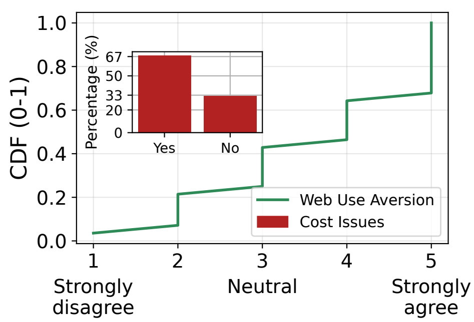


Figure 2: Barriers to Web access from speed and cost. Main plot: CDF of users avoiding sites due to slow internet. Inset: % of users limited by data costs.

average income levels. To assess these economic barriers and QoE in a rural setting, we conduct network performance measurements and a survey at the WebFM deployment site in Cameroon.

We select MTN [63], the best available mobile network provider in the area, and purchase its monthly data plan: 9.2 GB for $14. Following the methodology in [88], we use `mtr` [53], a tool for measuring latency and tracing network paths to examine routing changes and packet losses. Additionally, we employ `Speedtest CLI` [65], a command-line tool for measuring latency, download, and upload speeds. We also evaluate web performance using `Google Lighthouse` [45], an automated tool that provides key website performance metrics, including Speed Index (SI), First Contentful Paint (FCP), and Largest Contentful Paint (LCP). Given the data limits of our mobile internet plan, we conduct `mtr` and `Speedtest` measurements once every three hours. For Lighthouse evaluations, we rely on data collected in the wild as a response to real WebFM user requests (see Table 2).

Figure 1(a) shows the cumulative distribution function (CDF) of average round-trip time (RTT) towards popular content providers (Amazon, Facebook, and Google), and DNS operators (Cloudflare and Google). The figure shows RTTs higher than 100ms for most measurements and providers, with the exception of Google. This is due to Google's footprint in Africa [87], also confirmed by our path analysis. These high RTTs create a major bottleneck for web QoE, rendering modern, asset-heavy web pages slow or unresponsive.

Next, Figure 1(b) summarizes the speedtest analysis. With respect to the RTT to OOKLA servers (ping), the figure shows a similar trend as Figure 1(a). The figure further shows RTT under load, *i.e.,* while measuring both download and upload speeds, showing a 3.4x growth (from 200 to 680 ms, at the median) thus suggesting large buffer in use (a phenomenon typically called bufferbloat [42]). Despite these large buffers, users experience a median download speed of 2.6 Mbps and an upload speed of 0.69 Mbps—both drastically below the global average—at a mobile internet price comparable to developed nations ($1.5 per GB) [88]. At these speeds, web performance is extremely affected, as visualized in Figure 1(c). The median SI is at 9 seconds, FCP at 4.5 seconds and LCP at 6.9 seconds, significantly slower than the web vitals threshold (LCP < 2.5 seconds) recommended by Google [62].

The combination of high latencies and low download speeds can frustrate users and discourage regular internet use. Figure 2 presents responses to two survey questions assessing barriers to web access, drawn from our study with 30 participants in Cameroon. The main plot shows the CDF of web use aversion, based on responses to the question: *"To what extent do you agree with the following statement: 'I occasionally avoid visiting certain websites because my Internet is too slow to load them.'"* (1 = Strongly disagree, 5 = Strongly agree). Over 75% of the participants indicate either neutrality or agreement with avoiding websites due to slow internet speeds. The inset plot shows responses to the binary question: *"Is your web browsing experience often restricted by data costs?"* where 67% of users answered "Yes."

These results shed light on the state of web accessibility in low-income regions like Cameroon. Beyond slow speeds, high data costs severely limit both how often and how effectively users can engage with the internet, restricting access to information that many take for granted. Indeed, we also asked participants if they had heard of ChatGPT prior to participating in the experiment. 80% reported "No", revealing a broader lack of exposure to transformative (and popular) technologies simply because the current infrastructure never allowed these tools to reach them.

FM Radio Availability. FM radio still remains widely used in developing regions. In a study covering 39 countries in Africa, about 65% of adults reported listening to radio at least a few times per week, with no major difference in the rural-urban gap in radio access [18]. A recent case study in northern Ghana found FM radio to be the most reliable and trusted source of developmental information in rural communities, providing vital content on agriculture, education, and health in local languages [20].

Technical studies confirm that FM signals reliably cover large areas, though terrain can influence signal quality. In Nigeria, measurements around a 20 kW FM station showed stable reception up to

	WebFM	RDS	DARC	DirectBand	GPRS, 2G-4G	Starlink
Data Rate	10 kbps	0.73 kbps	16 kbps	12 kbps	9.6 kbps–100 Mbps	25-220 Mbps
Modulation	OFDM	BPSK	L-MSK	4-FSK	Various	QPSK/BPSK
Receiver	Mobile phone	FM receiver	Special receiver	SPOT devices	Mobile phone	Starlink terminal
Global Availability	Worldwide (FM)	Worldwide (FM)	Limited	Discontinued	Worldwide	87 countries
Interactivity	Limited via SMS	None	None	None	Full	Full
Monthly Cost	Free ($1 for SMS)	Free	Varies	Discontinued	$14 (Cameroon)	$40
Infrastructure Required	FM station + SMS network	FM station	FM subcarrier	DAB/FM infrastructure	Cellular network	Satellite network

Table 1: Comparison of wireless data delivery systems.

50 km, beyond which quality declined due to sandy or silty soil and obstructed line-of-sight [21]. A study from Nepal emphasized the importance of antenna height and placement in extending coverage, even in hilly regions with shadow zones between elevations [23]. In addition, research has revealed that frequency allocation for radio stations within the FM spectrum remains sparse across major cities worldwide [34, 67].

Building on the availability of FM radio connectivity, we next investigate the landscape of FM radio support in mobile phones. Specifically, we analyzed the prevalence of FM receivers in Android smartphones currently available on the market. To do this, we scraped the database of mobile phone specifications from GSMArena [48], identifying which models support FM radio and recording their release years, market availability, and prices. We then cross-referenced this information with global economic statistics [93] to assess the practical reach of FM-based data reception.

Despite a gradual decline in newer models, FM radio functionality remains prevalent—especially among Xiaomi and Vivo devices, where approximately 40% of current models still include FM support. In total, the top four Android brands (Samsung, Xiaomi, Vivo, and Oppo) [37] alone account for 571 FM-capable models, highlighting a significant and readily accessible user base for FM-based data delivery. We next evaluate the affordability of these 571 FM-capable phone models in low-income countries, using monthly gross national income (GNI) per capita as a benchmark [93]. Figure 3 shows the price distribution of these phones (blue line) alongside GNI thresholds (green dotted lines) for six developing countries: Pakistan, Cameroon, India, Bangladesh, Egypt, and Brazil. The figure shows that a substantial share of devices are within reach of average consumers, from a minimum of 30% in Pakistan up to 99.3% in Brazil. These results highlight strong market availability of FM-enabled phones at accessible price points in low and middle-income regions.

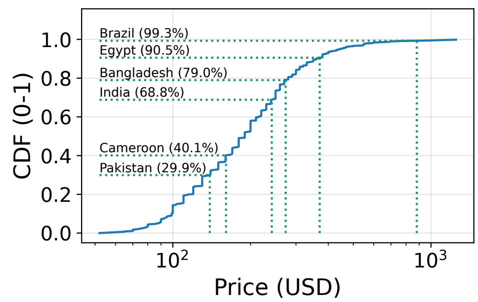


Figure 3: CDF of prices for Android phones equipped with FM radio. Green lines show monthly Gross National Income per capita in selected low- and middle-income countries.

Why Data Over Sound? Table 1 compares existing wireless data delivery technologies. RDS achieves only 1,187.5 bps raw throughput using biphase coding at 57 kHz [16]. The 10-bit error-correction overhead reduces effective throughput to 730 bps. As reported by Vigil et al. [90], RDS could only deliver 534 KB of content per hour, which is less than 12% of WebFM's hourly capacity, *i.e.,* 4.5 MB. Microsoft's DirectBand achieved approximately 12 kbps per tower but is discontinued. Data Radio Channel (DARC) improves transmission to 16 kbps using Level-controlled Minimum Shift Keying (L-MSK) on a 76 kHz subcarrier [84]. However, it requires specialized Large Scale Integration (LSI) chips for data reception, limiting its applicability to mobile devices. Unlike downlink-only services such as MSN Direct, WebFM targets commodity FM-capable Android phones and adds SMS uplink support for personalized URL and LLM requests, positioning it as an interactive access system rather than a fixed update service. Additionally, cellular data (GPRS, 2G-4G) and Starlink are expensive for users in developing regions like Cameroon as outlined earlier in this section.

At RDS's effective data rate of 730 bps, delivering even basic web content becomes infeasible. However, FM radio still provides excellent coverage of 30–60 km with superior building penetration. Instead of transmitting data via RDS, we achieve significantly higher data rates by encoding information as audio using OFDM—a multi-carrier modulation technique that partitions the available spectrum into multiple orthogonal narrow-band subcarriers. Using a configuration with 92 subcarriers and a center frequency of 9.2 kHz, we can encode data as sound and achieve transmission rates of 10 kbps. While this data rate is modest compared to cellular technologies (GPRS, 2G-4G) and satellite systems like Starlink, our approach leverages a widespread existing broadcast infrastructure without requiring specialized hardware modifications. This enables cost-effective deployment to millions of FM-capable devices in developing regions. Moreover, the broadcast nature eliminates per-user transmission costs while providing simultaneous delivery to unlimited receivers within coverage areas.

4 WebFM

Figure 4 shows how WebFM operates. Users A and B both have the WebFM app installed on their smartphones, equipped with chipsets featuring FM receivers. However, only User B's device supports SMS. The WebFM app runs on both smartphones with FM receivers

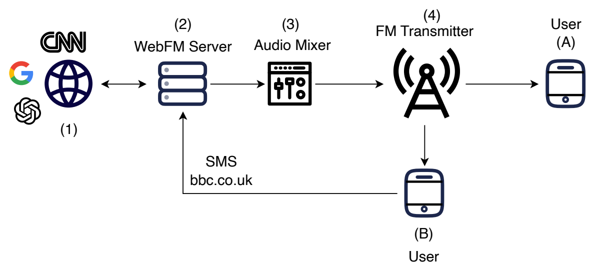


Figure 4: WebFM workflow.

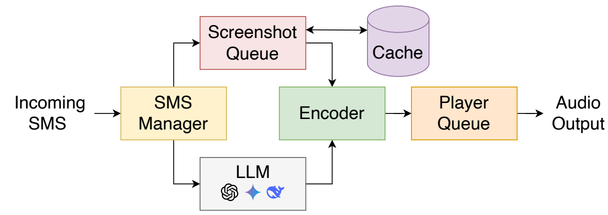


Figure 5: WebFM server architecture.

tuned to a specific frequency to decode incoming data-over-audio streams.

User B requests a webpage using the WebFM app, which sends an SMS containing the desired URL (e.g., `bbc.co.uk`) to the WebFM server. The server listens for incoming SMS messages, retrieves the webpage, captures a screenshot of the rendered page, and compresses it into a WebP image. The image is then encoded into sound and broadcasted via an FM transmitter. Both User A and User B receive the transmitted webpage on their smartphones. In the following, we describe each WebFM component which enables the above workflow; the implementation is open source at https://github.com/ayushpandeynp/webfm.

4.1 WebFM Server

Architecture. Figure 5 shows the architecture of the WebFM server. The server runs as a Docker [39] container on a computer located at a radio station. It consists of several key components we now describe. The SMS Manager handles incoming messages, while the Screenshot Queue processes webpage URLs in a First-Come, First-Served (FCFS) manner. A Cache stores recently requested URLs to avoid redundant processing. The Encoder converts responses into WebFM *file format* and encodes them to audio. Finally, the Player Queue manages the order in which the encoded audio files are played, also following the FCFS policy.

The SMS Manager listens for SMS messages using a USB mobile dongle. These messages are received on a phone number assigned to the SIM card inserted in the dongle. Messages sent by the WebFM app contain the sender's information in the headers and a payload formatted as: `<type> <body>` (e.g., `url https://nytimes.com`). When a new message is received, the server classifies it as either an LLM prompt or a webpage URL depending on the `<type>` identifier.

If the message contains a URL, it is added to the Screenshot Queue. The system checks whether this URL has been retrieved within the current transmission window using the Cache. If not, the Screenshot Queue utilizes Selenium [1] to load the page in Google Chrome with a mobile resolution of an iPhone SE device (375 × 667 pixels). Once the webpage is fully loaded, a full-page screenshot is taken and resized to a width of 320 pixels. We empirically selected 320 pixels as a sweet spot where both the content layout and text remain comfortably readable to the human eye. The screenshot is then encoded into a WebFM file format and subsequently converted into audio using the `Quiet` [11] library at the Encoder. The resulting audio is then added to the Player Queue. In case of cached requests, the cached response directly moves to the Player Queue. If the message contains an LLM prompt, the server makes an LLM inference API call to either a locally running LLM, or a cloud API (e.g., `OpenAI Chat Completions API` [66]). The response from the LLM is encoded into the WebFM file format containing structured metadata and payload. Finally, the server generates a corresponding audio using the `Quiet` library, which is then added to the Player Queue.

Encoding takes place in the Encoder, which consists of two steps: 1) encoding responses to a new file format, referred to as a *webfm file*; and 2) converting this webfm file into a waveform audio file (WAV). The resulting audio file is then added to the Player Queue.

WebFM File Format. WebFM uses a new file format that allows the decoder to easily distinguish specific parts of the transmission in absence of a continuous uplink. As illustrated in Figure 6, this format includes intermediate headers (such as "MDTA," "LNKS," "SDTA," etc.) that separate internal sections within the metadata and payload. This structure helps reconstruct content in cases when only parts of the payload are received properly. Each payload frame has a size of 500 bytes. To help distinguish the start point of each frame, each frame is prefixed with "C137," inspired by *Rick and Morty*'s original C-137 universe [73]. Furthermore, "C137" serves as a keepalive message, allowing the app to notify users that the server is online when transmissions are in progress.

Webpage Compression. Unlike encoding LLM interactions, where the payload is simply appended to metadata, encoding webpages presents a significant challenge due to the limited data rates achievable over audio transmission—typically only tens of kbps (see Section 2). According to [14], the average mobile webpage size is approximately 2 MB. Broadcasting such a page via WebFM could take tens of minutes. Moreover, devices experiencing poor RSSI may struggle to reconstruct the page, as critical web components, such as JavaScript, may fail in the presence of unrecoverable errors.

To address these challenges, we must: 1) significantly compress webpages, and 2) ensure resilience against noise. Web simplification approaches discussed in Section 2 remain vulnerable to noise and require extensive forward error correction (FEC), necessitating a system design that accounts for the worst-case receiver conditions.

Instead, we utilize a solution where performance degrades gracefully as a function of the receiver's RSSI, analogous to how audio quality deteriorates under poor reception. Inspired by [26, 27], which demonstrate how image quality over RDS degrades with RSSI, we opt to transmit images of rendered webpages rather than

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'M'      |      'D'      |      'T'      |      'A'      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Type (3 bytes, eg: "img", "url")        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Partition Sizes CSV                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'U'      |      'R'      |      'L'      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Request URL                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'I'      |      'D'      |      'E'      |      'N'      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Response Identifier (8 bytes)                 |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'W'      |             Page Width (3 bytes)              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'H'      |             Page Height (3 bytes)             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'E'      |      'O'      |      'M'      |      'D'      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'L'      |      'N'      |      'K'      |      'S'      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           JSON Array [{x, y, w, h, href}]                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'S'      |      'D'      |      'T'      |      'A'      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| 500-byte Frame Data Array:                                    |
|      'C'      |      '1'      |      '3'      |      '7'      |
| Frame Index + Partition Index + Data                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'I'      |      'D'      |      'E'      |      'N'      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Response Identifier (8 bytes)                 |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      'E'      |      'O'      |      'F'      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 6: WebFM file format.

raw web files (HTML/JS/CSS). This approach provides both compression and resilience: a 2 MB webpage can be compressed into a few hundred KB (see Appendix A), and images remain interpretable even if some pixels are lost.

Interactivity. Modern webpages enable user interaction via hyperlinks, menus, and search boxes, whereas images are inherently static. To introduce interactivity, [25] proposes *click maps*, which store `<x,y>` coordinates of interactive elements. We adopt this approach, allowing WebFM to notify the server (via SMS, if available) when a user clicks on a coordinate, retrieving the corresponding page if it is not already cached. Given WebFM's potentially slow network conditions–seconds for uplink and minutes for downlink–we limit interactivity to hyperlinks.

Image Format. Unlike [25], which requires lossless PNG for crowdsourced screenshot merging, WebFM utilizes *WebP* [3], a modern format offering superior compression. Webpages are captured as WebP images at 10% quality, significantly reducing file size while maintaining readability. Images are 320 pixels wide and up to 10,000 pixels tall, enabling users to *scroll* with minimal data overhead. To accommodate different screen sizes, images are resized using a scaling factor (`screen width / 320`), ensuring accurate click map coordinates.

Modulation. We use the `Quiet` library to modulate WebFM files as waveform audio files (WAV). Inspired by their "audible-7k-channel", we created a new modulation profile that uses OFDM with 92 subcarriers and a center frequency of 9.2 KHz, achieving a rate of 10 kbps.

Pushing. During low-usage periods, typically at night, WebFM *pushes* [83] popular webpages to its clients to ensure faster response times during peak hours. For each webpage, WebFM also pushes some internal links allowing users to seamlessly interact with a webpage without delay. However, pushing every hyperlink on a webpage would overwhelm the system's limited transmission bandwidth. To address this, we use a prioritization metric to rank internal links based on their importance. We derive this metric from a Prolific user study detailed in Section 5.

When a webpage enters the Screenshot Queue, WebFM computes this score for each hyperlink and selects the top three ranked links. These are added to a separate idle queue that is only activated when the server has no active transmissions. As WebFM primarily targets informative sites like news and blogs, this metric prioritizes pages with top headlines, larger images, and prominent font sizes, ensuring that key content is already available when users attempt to follow links. Additionally, it penalizes links that are vertically farther from the top of the page.

4.2 WebFM Client

At a minimum, WebFM users require a smartphone with a built-in FM radio receiver, serving as the *downlink*, along with a wired earphone to act as an antenna. Additionally, users who wish to send requests, such as retrieving a webpage or interacting with the LLM, need access to an SMS service for the *uplink*. On the software side, WebFM operates as a user-space application on a modified version of Android (see Figure 7). We detail the WebFM client in the following.

OS Integration. The WebFM client relies on FM radio hardware to receive data transmissions, but modern Android devices do not expose FM chip access to third-party applications. Default FM radio apps are shipped as system apps, integrated into the ROM and signed with privileged keys that allow hardware-level access. Apps like WebFM are unable to access FM audio without rooting the device and allowing superuser access, which is unrealistic for adoption and raises significant security concerns. As such, enabling FM-based decoding requires changes at the operating system level.

We build a proof-of-concept on LineageOS [59], a popular open-source Android distribution. We modify its default FM radio app to: 1) allow tuning the FM chip to a specific frequency, and 2) forward decoded audio streams to other apps like WebFM without needing root or elevated permissions. This is achieved with a `BroadcastReceiver` [19] in the WebFM app and a matching sender in the default FM radio app that transmits raw audio buffers in real time. This change allows WebFM to passively listen to FM broadcasts and decode data as it arrives. We verified this approach by flashing our customized LineageOS build onto devices that use Qualcomm Snapdragon chipsets, including Xiaomi Redmi Go. We observe that as long as FM radio drivers are available, this approach can be extended to any other chipset. Manufacturers would implement this change when building the stock operating system for devices they ship with FM radio support (see Section 4.3).

User Interface. Figure 8 shows the WebFM app's user interface (UI), which consists of three sections: Browser, ChatGPT and Knowledge Hub. The Browser section mimics modern web browsers. It features a search bar at the top, allowing users to request URLs. A list of requested URLs is also displayed, and once received, they appear as "ready to view." The ChatGPT section provides a chat-like interface where users can interact with ChatGPT by asking questions and receiving responses. Since FM radio operates as a broadcast system, all devices tuned to the same frequency receive the same content, even if it wasn't specifically requested by those

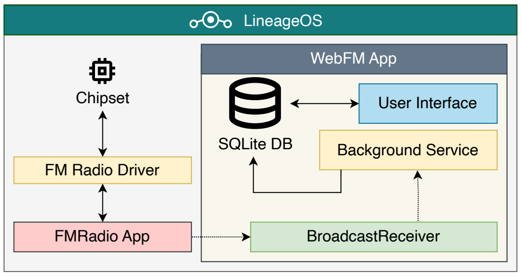


Figure 7: WebFM client architecture.

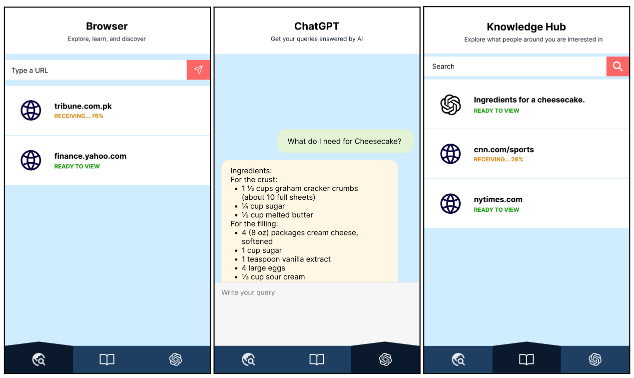


Figure 8: WebFM's user interface.

devices. This broadcast nature of FM radio is leveraged by WebFM with its Knowledge Hub section. In this section, users can access a list of webpages and ChatGPT responses that are popular within their region, allowing them to discover trending content shared by others nearby.

Background Service. In addition to the UI, the WebFM app runs a background service that performs two key functions: 1) continuously listening to FM radio audio streams at a specific frequency, and 2) decoding transmissions when a WebFM-encoded signal is detected. The background service listens to bytes broadcasted by the default FM radio app using a `BroadcastReceiver`. To decode WebFM-encoded transmissions, it uses a modified version of `Quiet`'s Android library [4], which by default decodes audio from the device's microphone. We modified the library to accept bytes retrieved from the phone's FM radio app and decode them using the same modulation profile described in Section 4.1.

This background service operates independently of the user interface and automatically stores all incoming transmissions in an SQLite database. To optimize storage, any unaccessed content is automatically deleted after one day. Outside operational hours, the FM radio is turned off to conserve energy. However, when operating in "on-demand" mode (where the WebFM server immediately responds to user requests), FM radio remains on all the time, except when users explicitly switch stations to listen to regular broadcasts.

Error Correction. We utilize crc32 checksums per frame to detect errors. Furthermore, an inner FEC scheme (v29) and an outer FEC scheme (rs8) are used to correct transmission errors. For completely lost frames, the WebFM receiver applies nearest-neighbor pixel interpolation [75], replacing missing pixels with the value of their adjacent left pixel given that the webpage consists mostly of text read from left to right.

In the literature, better-performing techniques exist to recover missing pixels in images [30, 71, 82], leveraging deep neural networks to learn patterns and structures of the image or utilizing sparsity and gradients in the data to fill in the missing regions. These techniques are both memory- and CPU-intensive, far beyond what a low-end mobile device can support today. Thus, we adopt a lightweight approach proposed and benchmarked by [68] that provides consistently high content readability scores even at a 20% pixel loss rate.

4.3 Discussion

Rollout. In the short term, governments, NGOs, etc. can distribute WebFM by pre-installing a modified version of the LineageOS ROM on existing Android devices (detailed in Section 4.2). In the long term, smartphone manufacturers can natively support WebFM by bundling a modified FM radio app that exposes audio streams to other applications. This would make FM-based services broadly accessible out of the box, without requiring root access.

Incentives and Economic Viability. FM broadcasting costs remain constant regardless of the number of listeners, making WebFM an attractive approach to attract new users, who benefit by gaining offline access to a streamlined version of the Web in areas lacking cellular coverage. At 10 kbps transmission rate, approximately 1 GB of raw data can be broadcast per month assuming a 7 hr transmission window overnight. With WebFM's 10× web content compression (see Appendix A), this translates to roughly 10 GB of usable content per month for less than $1. In comparison, MTN's $14/month subscription provides 9.2 GB of data and it is inaccessible to many, given that around 38% Cameroonians (~10–11 million people) live below the national poverty line [61], and 23% (~6.3 million) live under extreme international poverty [51].

Providers can monetize the service in several ways: 1) by bundling with an SMS service, 2) by charging subscription fees, or 3) through ad-based free access. In the subscription model, users pay a monthly fee to access premium features. For example, subscribers can be given priority when requesting content from the curated list of websites they manage via SMS. In the ad-supported model, users do not pay any fees, but a portion of each webpage screenshot they receive is reserved for embedded advertisements.

Privacy and Security. WebFM resembles acceleration services like Google AMP [44] and WebLight [9], which modify webpage content before delivery. These intermediaries have visibility into user browsing patterns, raising potential privacy concerns. However, WebFM downlink-only users are not required to identify themselves to the broadcaster, since they only passively receive data requested either by nearby users or sent periodically by the WebFM server. This provides receiver-side privacy for popular pages, but it does not provide full anonymity: a passive FM-broadcast observer could still link rare content to a specific user when the transmitted content is sufficiently unique. For rare or sensitive pages and LLM queries, WebFM can selectively use per-user encryption. During enrollment, the client stores the broadcaster's public signing key and the server stores the client's public encryption key. Transmissions remain broadcast, but private content can be decrypted only by the intended client. Popular pages with low inference risk can remain unencrypted to preserve broadcast efficiency.

WebFM uplink users face additional privacy risks because SMS requests can expose both the requested webpage and the user identity associated with the request. In particular, a request sent through the SMS uplink may link the user's phone number to the requested webpage, unless a non-colluding proxy service is added in place [85, 89]. This risk is compounded by the fact that, at the carrier level, SMS messages are stored and transmitted in plain text. In principle, we can mitigate this by encrypting SMS payloads with a public key. This is similar to how HTTPS bootstraps trust through public/private key exchanges. To reduce exposure during the initial key exchange, WebFM can rely on standard mechanisms such as server certificates [36] and certificate pinning [41], ensuring that the client only accepts the legitimate WebFM server's key.

Content spoofing presents significant risk in broadcast systems like WebFM. Attackers could intercept the transmission frequency and rebroadcast modified content using stronger FM signals. To mitigate this, we introduce an optional signature that: 1) introduces minimal overhead to WebFM 500-byte frames, and 2) operates efficiently on resource-constrained mobile processors. Ed25519 [24], an elliptic curve digital signature algorithm widely adopted in TLS 1.3, SSH, Tor [28], addresses these requirements with 64-byte signature overhead. Our lab experiments show that adding Ed25519 signatures reduces data rate to 8.6 kbps. This drop may be worthwhile in scenarios where data integrity is critical, as it ensures broadcasts remain resistant to spoofing.

Limitations. WebFM does not enable access to login-restricted content, such as online banking or social media accounts. This is unfeasible for downlink-only users (*i.e.,* no SMS support); for uplink users, it would involve sharing login credentials with the WebFM server which is a significant privacy risk. Moreover, because content is broadcasted, any personalized information (like account details) would be exposed to anyone within range, further compromising privacy.

Next, WebFM lacks support for video which is a major part of modern web usage, e.g., streaming, news, and social media. WebFM's limited bandwidth makes video streaming infeasible. Instead, video content is replaced with static, non-interactive thumbnails. Likewise, advanced features driven by JavaScript or CSS are not supported, as WebFM only transmits simplified, pre-rendered versions of webpages.

5 Benchmarking

Signal Strength and WebFM Performance. We place five Xiaomi Redmi Go phones, each with the WebFM app installed, at fixed distances of 5–50 m from a 0.5 W FM transmitter to artificially create diversity in RSSI. The experiment is conducted indoors in a crowded university building, where signal variability arises from path loss, structural obstruction, multipath propagation, and intermittent body blockage caused by the movement of students and staff. The phones are positioned at tabletop height, with three devices placed on the same floor as the transmitter and the remaining two placed one floor below, further diversifying the received signal conditions. All devices are kept fully powered meanwhile 5,000 randomly-selected webpages from the Tranco [56] list are broadcasted over FM at a frequency of 91.5 MHz, so that they are concurrently received by the testing devices while emulating varying RSSI. Figure 9 shows the CDF of loss rates across all transmissions and its inset plot presents loss percentage as a function of RSSI range. The main CDF shows that over 95% of transmissions experienced loss rates below 10%, well within the tolerance threshold for content readability [68]. Furthermore, high loss percentages (for the remaining 5% of transmissions) are largely confined to relatively poor signal conditions, particularly in the RSSI range of −70 to −60 dBm. As RSSI improves, the loss percentage rapidly declines and stabilizes near zero.

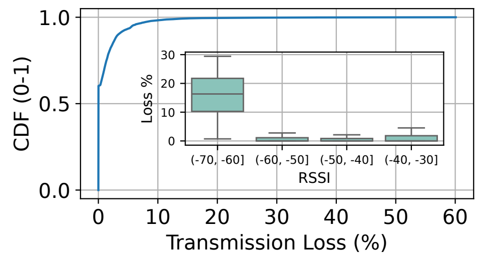


Figure 9: CDF of loss percentage under ideal conditions. Inset shows loss % vs RSSI.

Pixel Interpolation. [68] evaluates the impact of visual loss and pixel interpolation on perceived content clarity and text readability using feedback from 151 Pakistani university students across 50 test webpages. Their results show that even at a 20% pixel loss rate – which is rare in WebFM as shown in Figure 9 – users reported a median content clarity score of 7 out of 10, indicating a generally clear understanding of the page. While text readability was more affected, it remained acceptable at a loss rate of 20%.

Pushing Metric. We conduct a user study on Prolific [15] to evaluate the likelihood of hyperlink clicks based on visual features of each link. Specifically, we examine the area covered by the link in a webpage screenshot (width · height) and its vertical position on the page (y-position). We randomly sample 100 webpages from the Tranco list. For each page, we generate mobile screenshots along with the bounding-box coordinates (x, y, w, h) of every hyperlink. We then create an interactive webpage where participants are asked to click on the link they would "naturally" choose to visit next.

A total of 100 participants take part in the study, each interacting with 10 different pages. The pages are distributed so that each webpage is evaluated by ten users, resulting in 1,000 total page interactions. Using the data collected from our study, we train a logistic regression model to learn the relative importance of a hyperlink's area and vertical position in predicting click likelihood. The model fits a weighted linear combination of these features to estimate the probability of a link being clicked. The resulting scoring function is: $score = 0.68 \cdot w \cdot h - 0.32 \cdot y$, where w, h, and y denote the width, height, and vertical distance from the top of the page. The negative weight on y-position reflects that links appearing closer to

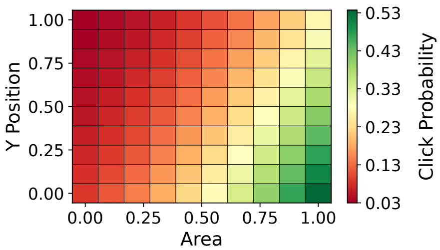


Figure 10: Likelihood of clicking on a hyperlink in a webpage given its area and y position. Data from 1,000 page interactions from 100 users across 100 pages.

Figure 11: FM radio station in Cameroon, and WebFM server located inside the station.

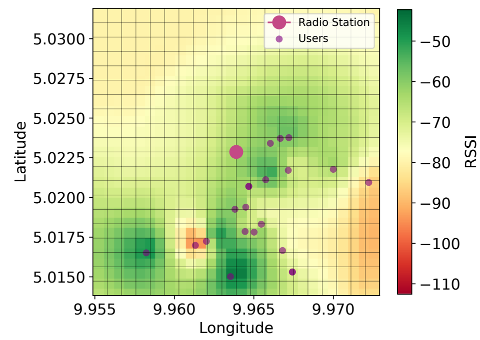


Figure 12: Map of the 30 WebFM users with their interpolated RSSI values. Each cell in the grid is 100 meters in scale. Note that some users are overlapping.

the top (*i.e.,* with lower y-values) are more likely to be clicked. We use this scoring function as our prioritization metric for pushing.

Figure 10 shows the likelihood of clicking a hyperlink based on its area and vertical position on the page. The heatmap shows that links with larger areas and located closer to the top (*i.e.,* lower y-values) are more likely to be clicked. This trend is visible in the gradient transition from red (low click probability) to green (high click probability), moving from the bottom-left to the top-right of the plot. Although the maximum observed click probability is only 0.53, the relatively higher likelihood still offers a useful signal for prioritization. Since the server remains underutilized during idle periods, pushing these links can still improve user experience with minimal additional cost.

6 Deployment

This section outlines the WebFM deployment at a live FM radio station in Cameroon. We selected this location given its low internet penetration rate comparable to low-income regions (58.1% of Cameroon's population is offline as of 2025 [38]).

Methodology. We start by signing an agreement with an FM radio station "La Voix de la Diversité" in Baré Bakem, Cameroon to allow the WebFM server to transmit content from 10 PM to 5 AM daily for six weeks. This overnight window was the only available airtime, as the station's daytime schedule was reserved for regular programming. Such opportunistic use of off-peak radio hours represents the most feasible adoption path for WebFM in the near term. In the future, we envision dedicated FM channels operating full-time for data broadcasting.

We recruited 30 Cameroonians to experiment with WebFM during this period, *i.e.,* request webpages and ask questions to ChatGPT. Study participants were given a Xiaomi Redmi Go phone (featuring Qualcomm Snapdragon 425 processor and 1 GB RAM) flashed with the modified version of LineageOS, and WebFM app pre-installed. To send requests, each phone had a SIM card with an unlimited SMS bundle.

As shown in Figure 11, the WebFM server was set up at the FM radio station using a MacBook Air with 8 GB of RAM, running the WebFM Docker container. We used Huawei's E8372h-320 LTE/4G USB Mobile WiFi Dongle [50] to interface with the SIM card via `huawei-lte-api` [77] and receive incoming SMS messages. The WebFM app was programmed to send SMS messages to the number associated with the SIM card used by the dongle. For internet access, we used a mobile data subscription from MTN Cameroon (see Section 3 for details on plan and connection quality).

Data Collection. Table 2 outlines our deployment. A total of 30 participants were recruited, divided into two sequential batches of 15 participants each. The study spanned 6 weeks in total, with each batch participating for 3 weeks. As an incentive to the participants, we offered USD 2 per person per day. Before the study, participants signed a consent form and were allowed to withdraw from the study at any time. The participants were then given an introduction on how to use the WebFM app; further, an institutional review board (IRB) approval was granted to conduct the study. The authors who conducted the study are CITI [2] certified. No sensitive or personal information of the participants was collected, except for their name and phone number to contact them and disburse the incentive money at the end of the experiment.

Study participants were allowed to make up to 10 requests per day–this included both webpage URL requests and GPT queries. Participants were allowed to make requests at anytime during the day; however, responses were transmitted during the transmission window of 7 hours (10 PM to 5 AM). Over 3 weeks, participants made 1,737 URL requests and 2,936 GPT queries in total. The median number of requests per user was 160, with 96 GPT queries and 64 URL requests.

Deployment Challenges. We encountered several challenges during WebFM deployment. Initially, airport security confiscated 10 mobile phones intended for participants, significantly reducing the number of available devices. Only 15 phones ultimately reached

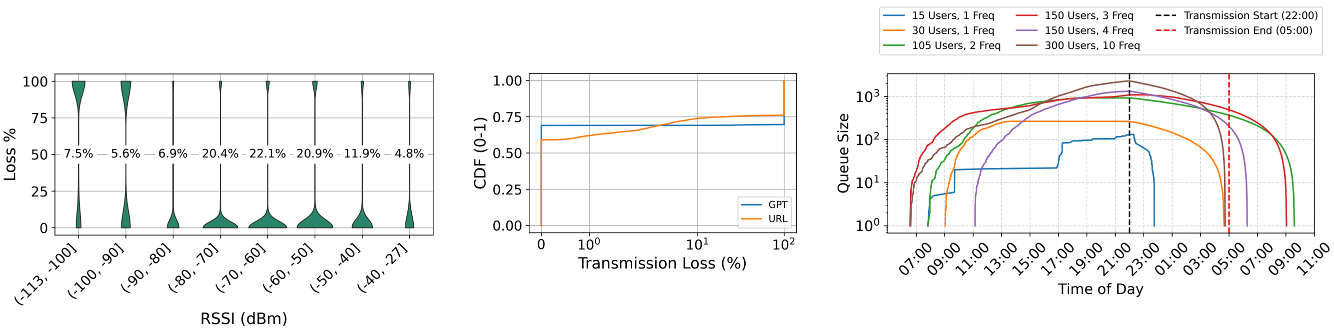


Figure 13: WebFM evaluation from the wild. (a) Loss % as a function of RSSI range. (b) CDF of loss % for GPT and URL transmissions. (c) Queue size for *actual* (15 users, 1 frequency), and *simulated* (30–300 users, 1–10 frequencies) loads assuming the busiest day of WebFM deployment.

Property	Description
Location	Cameroon
Participants	30
Number of batches	2 (15 participants per batch)
Duration of study	6 weeks
Duration per batch	3 weeks
Transmission window	10 PM to 5 AM daily
Daily request quota	10 (GPT + URL)
Total requests	1,737 (URL) and 2,936 (GPT)
Median requests per user	Total: 160, GPT: 96, URL: 64 (in 3 weeks)

Table 2: Summary of WebFM deployment.

Cameroon, forcing us to conduct the experiment in two separate batches. Securing reliable internet connectivity for the WebFM server also proved challenging: even the best available mobile internet plan from MTN was unstable (as discussed in Section 3), occasionally unavailable for entire days, and affected by significant latency. Compounding these issues, the village where we deployed frequently experienced electricity outages—lasting up to 8 hours and often overlapping with the transmission window—completely disrupting FM radio transmissions. During these outages, users continued to request content via SMS but were unable to receive any responses.

Operational issues further complicated the deployment. In the first batch, the WebFM app sent acknowledgment messages (ACKs) for all received transmissions, enabled by unlimited SMS bundles purchased for each participant. However, the resulting high volume of SMS traffic quickly raised suspicion with the mobile operator, leading to the blocking of all deployed SIM cards. Consequently, we had to disable the ACK mechanism, leaving us without real-time operational feedback or heartbeat signals from participants' phones. Additionally, the success of FM radio transmissions critically depended on the radio station staff accurately switching to WebFM broadcasts at exactly 10 PM each night. This switch-over, however, was inconsistent, resulting in multiple days without any transmissions. We also discovered that the transmitter's output volume needed to be set to 100% to achieve better range and improve transmission quality, but maintaining this setting consistently proved challenging for the radio staff as well. Finally, the effectiveness of the WebFM system relied heavily on participants regularly charging their phones and keeping earphones connected at all times, as the earphones served as antennas for receiving broadcasts.

7 Results

RSSI and Loss Analysis. Figure 12 shows the spatial distribution of RSSI measurements across a 100m-resolution grid around the FM radio station. Using ordinary kriging [91], we interpolate user-collected GPS-tagged signal data to generate a continuous RSSI map. While the radio station is centrally located, the strongest signal regions are notably offset to the northeast, with two additional users registering high RSSI values at distances of approximately 900–950 m (one directly to the south and another to the southwest). The northeastward bias in signal strength likely results from the antenna's physical orientation or directional configuration, while the isolated strong-signal detections at longer distances suggest favorable line-of-sight conditions. This corroborates prior observations [23] that suggest that, contrary to the expected radial decay in signal strength with distance, close proximity to the radio station does not guarantee strong signal strength and other factors such as antenna height and placement can have a significant influence on signal quality.

Figure 13(a) shows transmission loss percentages across RSSI ranges, with each violin's width representing the proportion of total transmissions in that range (e.g., 22.1% of all transmissions occurred between -70 and -60 dBm). In poor signal conditions below -90 dBm, the loss rate is frequently near 100%. For stronger signals in the -80 to -50 dBm range (accounting for above 60% of total transmissions), data points are more concentrated below 20%, and instances of 100% loss are rare. Overall, we observe a clear trend: better signal strength (*i.e.,* higher RSSI) is associated with reduced transmission loss and a high rate of successful transmissions. This corroborates, in the wild, the observations from our benchmarking under controlled settings (see Figure 9).

To investigate per user performance, Figure 14(a) shows boxplots of RSSI values per participant alongside their completion rates for both GPT and URL requests. Users with stronger median RSSI values (e.g., above −70 dBm) consistently achieve high completion rates across both content types. For instance, users 15–17, 21–28, and 30 report more than 80% completion rates, indicating that strong and

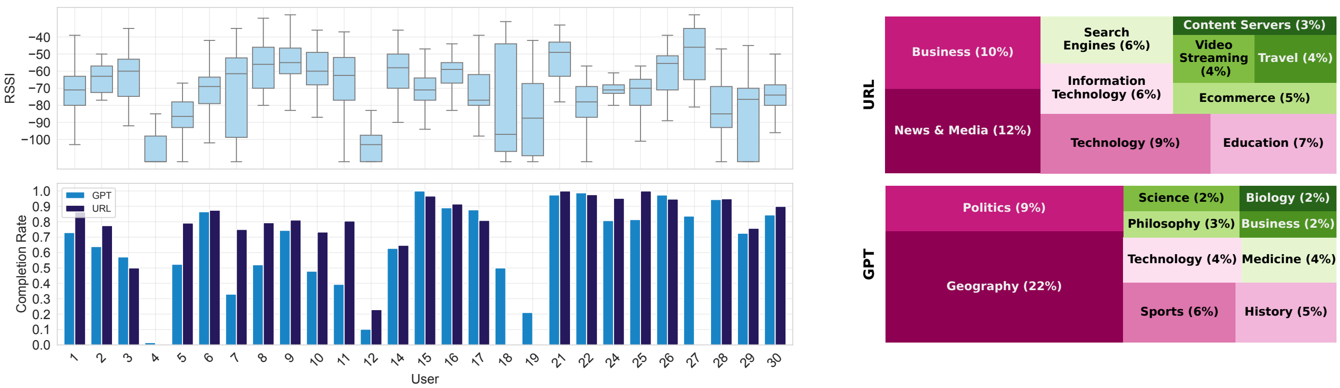


Figure 14: (a) RSSI distribution per user with their request completion rates. Users 13, 20, and 23 were removed as their devices appeared faulty. (b) Top 10 categories of URL and GPT content requested by WebFM users.

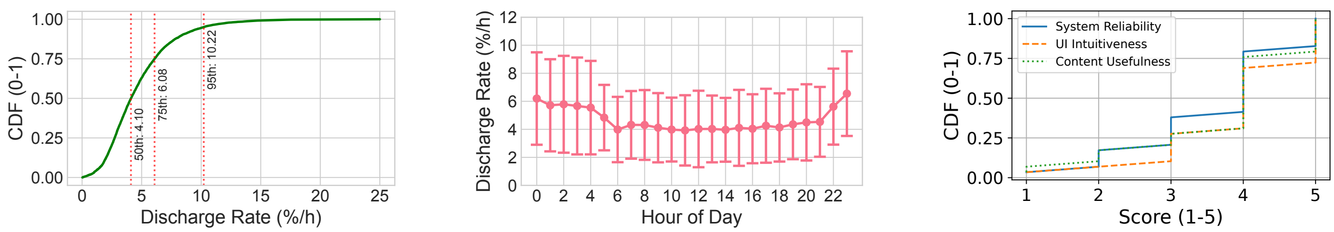


Figure 15: (a) Battery discharge rate in % for phones running the WebFM app. (b) Battery discharge rate pattern by hour of day. 10 PM to 5 AM reflects the transmission window. (c) User experience of WebFM on system reliability, UI intuitiveness, and content usefulness.

stable signal conditions are sufficient for reliable content delivery. In contrast, users with lower and unstable signal quality, particularly those with median RSSI fluctuating below −90 dBm (e.g., users 4, 12, and 19), show drastically reduced completion rates, often below 30%.

Due to their smaller sizes, GPT responses have shorter broadcast duration than webpages. However, they are more susceptible to failures from partial frame loss. A single 500-byte frame drop can disrupt an entire GPT message, whereas similar loss in a webpage screenshot has limited impact (see Figure 17). As a result, webpages achieve higher completion rates. This can be observed for users 7, 8, 10, and 11, where URL completions are consistently higher than GPT under similar signal conditions. This trend is also reflected in Figure 13(b), which shows the CDF of loss rates observed for the two types of content—GPT responses and URL transmissions. URL transmissions (orange line) appear more loss-resilient than GPT (blue line) due to their shorter duration and stricter tolerance to partial frame loss.

Scalability. Figure 13(c) shows the evolution of the transmission queue over the course of a day. We select the busiest day to understand transmission trends within the transmission window (marked by the vertical dashed lines at 22:00 and 05:00). The blue trace represents the real-world deployment of WebFM, with 15 users receiving content on a single frequency. Although each user requested 10 pages, on average, the queue size peaked at 103 items, indicating that roughly 30% of transmissions were served from the cache. The sharp spike at 09:30 corresponds to a scheduled "push" of pre-selected news pages during an otherwise idle period. All other curves represent First-Come, First-Served (FCFS) queue simulations, scaling this baseline traffic to heavier loads. For example, with 30 users on a single FM frequency (orange), the queue peaks at around 200 items but is still fully transmitted within the 7-hour transmission window. This suggests that one frequency can support up to 30 active users with similar queuing patterns. When capacity increases to two frequencies and 105 users (green), the peak backlog rises to 800 items, half of which could not be transmitted before 05:00. Similar patterns are observed for 150 users with three and four frequencies, respectively. However, a configuration of 300 users with ten frequencies appears sufficient to fully transmit the queue within the transmission window.

Content Analysis. Figure 14(b) presents a treemap of the top 10 content categories for both GPT queries and URL requests made by WebFM users. GPT queries are classified using Meta's Llama 3 model [46], while domains in URL requests are categorized using Cloudflare's Domain Intelligence API [35] following the methodology in [76]. Among GPT queries, Geography dominates with 22% of all requests, followed by Politics (9%), Sports (6%), History (5%), Technology (4%), and Medicine (4%). Smaller but still notable portions of queries are related to Philosophy, Science, Business, and Chemistry. For URL requests, News & Media (12%) and Business (10%) are the most common, followed by Technology and Education. Users also accessed sites in categories such as Search Engines, Ecommerce, Travel, and Video Streaming. Content from these latter categories is less likely to be useful given WebFM's current limitation of only supporting page screenshots with limited (and slow) hyperlink interactivity.

Battery Consumption. Figure 15(a) presents the CDF of battery discharge rates measured on Xiaomi Redmi Go devices (3,000 mAh lithium-ion battery), running the WebFM application when not connected to external power. The median discharge rate of 4.10% per hour indicates that typical devices can operate for almost 24 hours on a single full charge. The 95th percentile at 10.22% per hour is rare, but even the highest consumers can maintain about 10 hours without needing to recharge. Figure 15(b) reveals the hourly discharge pattern, with error bars indicating variability across timings. During the transmission window, *i.e.,* 10 PM to 5 AM, discharge rates consistently increase by approximately 2%, as devices are primarily passively listening to FM radio and processing incoming data. Noticeably, at the start of the window, we observe a spike reaching up to 6.5% and a steady decrease in consumption as the transmission queue becomes shorter after midnight. The flat consumption outside the transmission window validates that the application maintains minimal background activity during non-operational hours, as the FM radio is turned off and no processing is carried out.

User Experience. At the end of the experiment, we asked participants to complete an exit survey assessing their experience with WebFM across three dimensions. Each question was rated on a 5-point likert scale, with varying response ranges depending on the aspect evaluated. For system reliability, users responded to: *"How reliable was the content received through the WebFM app?"* (1 = Not reliable at all, 5 = Very reliable). For UI intuitiveness, they answered: *"How intuitive is the user interface of the WebFM app?"* (1 = Not intuitive at all, 5 = Very intuitive). Lastly, for content relevance, users rated: *"How useful was the content you received from the system?"* (1 = Not useful at all, 5 = Very useful).

Figure 15(c) shows the CDF of responses to the survey questions. Overall, user feedback was positive: 72% of participants rated the WebFM app as "intuitive" and the content as "useful" (with scores of 4 or 5). In contrast, system reliability received relatively lower ratings, with only 62% of users selecting 4 or 5. This is expected, as some participants experienced fluctuating or low RSSI values, leading to reduced request completion rates, as discussed earlier in Figure 14(a).

8 Conclusion

Despite decades of progress in global connectivity, billions of people around the world remain offline—not due to a lack of infrastructure, but because of persistent affordability barriers. Access to the internet continues to be out of reach for many, especially in resource-constrained regions where even low-cost mobile data can be prohibitively expensive. In this context, WebFM introduces a novel, ultra-low-cost approach to narrowing the digital divide by leveraging FM radio—a ubiquitous, inexpensive, and underutilized medium—to deliver essential web content and large language model (LLM)-based interactions without requiring an internet connection. By combining a full-system design, seamless integration with Android FM tuners, and deployment in real-world settings such as Cameroon, we demonstrate that WebFM can reliably transmit simplified web content and AI-generated responses in a way that is accessible, scalable, and resilient. Our work showcases the untapped potential of repurposing existing broadcast infrastructure to extend digital access to underserved populations, offering a practical path forward for connecting the unconnected and promoting more equitable access to knowledge and services across the globe.

9 Ethics

This work does not raise any ethical issues, as the study was approved by the IRB and all authors conducting the research were CITI [2] certified.

References

[1] 2017. *SeleniumHQ Browser Automation.* http://www.seleniumhq.org/about/
[2] 2019. CITI Program - Collaborative Institutional Training Initiative. www.citiprogram.org. Accessed: 2022-05-18.
[3] 2023. An image format for the Web. https://developers.google.com/speed/webp.
[4] 2023. Android app built with Quiet which allows o pass data through the speakers of an Android device. https://github.com/quiet/org.quietmodem.Quiet.
[5] 2023. AudioQR. https://github.com/jamesonrader/AudioQR.
[6] 2023. Expanding internet connectivity with stratospheric balloons. https://x.company/projects/loon/.
[7] 2023. GGwave. https://github.com/ggerganov/ggwave.
[8] 2023. High altitude connectivity: The next chapter. https://engineering.fb.com/2018/06/27/connectivity/high-altitude-connectivity-the-next-chapter/.
[9] 2023. Load mobile pages faster with Web Light. https://support.google.com/websearch/answer/9836344?hl=en.
[10] 2023. MSN Direct. https://en.wikipedia.org/wiki/MSN_Direct.
[11] 2023. Quiet modem project. https://github.com/quiet/quiet.
[12] 2023. Safari Reader Mode. https://support.apple.com/guide/safari/hide-ads-when-reading-sfri32632/mac.
[13] 2023. Starlink. https://starlink.com/.
[14] 2023. Web Almanac. https://almanac.httparchive.org/en/2022/.
[15] 2025. Prolific | Quickly find research participants you can trust. https://www.prolific.com/.
[16] 2025. Radio Data System. https://en.wikipedia.org/wiki/Radio_Data_System.
[17] 2025. Taara: A Google X Moonshot. https://x.company/projects/taara/ Accessed: 2025-03-19.
[18] Afrobarometer. 2024. Africa's shifting media landscapes: Digital media use grows, but so do demographic divides. https://www.afrobarometer.org/wp-content/uploads/2024/04/AD800-PAP14-Digital-media-use-grows-across-Africa%5ELJ-but-so-do-demographic-divides-Afrobarometer-29april24.pdf
[19] Android Developers. 2025. BroadcastReceiver. https://developer.android.com/develop/background-work/background-tasks/broadcasts Accessed: 2025-04-10.
[20] Osman Antwi-Boateng, Muhammed Danladi Musa, and Mu-Azu Iddirisu Andani and. 2023. Audience listenership of FM radio: A case study of rural development in Northern Ghana. *Cogent Arts & Humanities* 10, 1 (2023), 2184750. https://doi.org/10.1080/23311983.2023.2184750 arXiv:https://doi.org/10.1080/23311983.2023.2184750
[21] Evans C. Ashigwuike, Ale Felix, and Farouq E. Shaibu. 2019. The Impact of Soil Texture on Path Loss Modelling of an FM Signal Using Diffraction Technique: A Case Study of Prime FM Radio Nigeria. *European Journal of Engineering Research and Science* 4, 4 (April 2019), 56–63. https://doi.org/10.24018/ejers.2019.4.4.1231
[22] Yang Bai, Jian Liu, Li Lu, Yilin Yang, Yingying Chen, and Jiadi Yu. 2020. BatComm: enabling inaudible acoustic communication with high-throughput for mobile devices. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems.* 205–217.
[23] Tina Baidar. 2017. Signal Coverage Mapping of Local Radios. *Journal on Geoinformatics, Nepal* (2017). Accessed via Nepal Journals Online.
[24] Daniel J. Bernstein, Niels Duif, Tanja Lange, Peter Schwabe, and Bo-Yin Yang. 2012. High-speed high-security signatures. *Journal of Cryptographic Engineering* 2, 2 (2012), 77–89. https://doi.org/10.1007/s13389-012-0027-1
[25] Ketan Bhardwaj, Ada Gavrilovska, Moritz Steiner, Martin Flack, and Stephen Ludin. 2018. DRIVESHAFT: Improving Perceived Mobile Web Performance. *arXiv preprint arXiv:1809.09292* (2018).
[26] Radu Gabriel Bozomitu and Florin Doru Hutu. 2023. Drivers' Warning Application Through Personalized DSSS-CDMA Data Transmission by Using the FM Radio Broadcasting Infrastructure. *IEEE Access* 11 (2023), 11711–11731.
[27] Radu Gabriel Bozomitu and Ştefan Corneliu Stoica. 2022. A Robust Radiocommunication System for FM Transmission Based on Software Defined Radio Technology. In *2022 IEEE 28th International Symposium for Design and Technology in Electronic Packaging (SIITME).* IEEE, 78–81.

[28] Jacqueline Brendel, Cas Cremers, Dennis Jackson, and Mang Zhao. 2021. The Provable Security of Ed25519: Theory and Practice. In *2021 IEEE Symposium on Security and Privacy (SP)*. 1659–1676. https://doi.org/10.1109/SP40001.2021.00042
[29] Michael Butkiewicz, Daimeng Wang, Zhe Wu, Harsha V. Madhyastha, and Vyas Sekar. 2015. Klotski: Reprioritizing Web Content to Improve User Experience on Mobile Devices. In *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI '15)*. USENIX Association, Oakland, CA, USA. https://www.usenix.org/conference/nsdi15/technical-sessions/presentation/butkiewicz Open Access.
[30] Xintao Chai, Hanming Gu, Feng Li, Hongyou Duan, Xiaobo Hu, and Kai Lin. 2020. Deep learning for irregularly and regularly missing data reconstruction. *Scientific Reports* 10, 1 (24 Feb 2020), 3302. https://doi.org/10.1038/s41598-020-59801-x
[31] Moumena Chaqfeh, Russell Coke, Jacinta Hu, Waleed Hashmi, Lakshmi Subramanian, Talal Rahwan, and Yasir Zaki. 2022. Jsanalyzer: A web developer tool for simplifying mobile web pages through non-critical javascript elimination. *ACM Transactions on the Web* 16, 4 (2022), 1–31.
[32] Moumena Chaqfeh, Muhammad Haseeb, Waleed Hashmi, Patrick Inshuti, Manesha Ramesh, Matteo Varvello, Lakshmi Subramanian, Fareed Zaffar, and Yasir Zaki. 2022. To Block or Not to Block: Accelerating Mobile Web Pages On-The-Fly Through JavaScript Classification. In *Proceedings of the 12th International Conference on Information and Communication Technologies and Development, ICTD 2022, Seattle, USA, June 27-29, 2022 (accepted)*.
[33] Moumena Chaqfeh, Yasir Zaki, Jacinta Hu, and Lakshmi Subramanian. 2020. JSCleaner: De-Cluttering Mobile Webpages Through JavaScript Cleanup. In *Proceedings of The Web Conference 2020*. 763–773.
[34] Rajashekar Reddy Chinthalapani, Dhairya Jigar Shah, Nobel Ang, and Ambuj Varshney. 2025. AudioCast: Enabling Ubiquitous Connectivity for Embedded Systems through Audio-broadcasting Low-power Tags. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 9, 2, Article 27 (June 2025), 32 pages. https://doi.org/10.1145/3729471
[35] Cloudflare, Inc. 2025. *Threat Intelligence APIs - Security Center*. https://developers.cloudflare.com/security-center/intel-apis/ Accessed: 2025-03-22.
[36] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk. 2008. *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. RFC 5280. Internet Engineering Task Force. https://datatracker.ietf.org/doc/html/rfc5280
[37] Counterpoint Research. 2025. Global Smartphone Market Share: Quarterly. (2025). https://www.counterpointresearch.com/insights/global-smartphone-share/
[38] DataReportal. 2025. Digital 2025: Cameroon. https://datareportal.com/reports/digital-2025-cameroon Accessed: 2025-03-26.
[39] Inc. Docker. 2024. Docker: Open platform for developing, shipping, and running applications. https://www.docker.com/
[40] Emmanuel Engelhart and Renaud Gaudin. 2025. *Kiwix*. https://kiwix.org/en/
[41] C. Evans, C. Palmer, and R. Sleevi. 2015. *Public Key Pinning Extension for HTTP*. RFC 7469. Internet Engineering Task Force. https://datatracker.ietf.org/doc/html/rfc7469
[42] Jim Gettys and Kathleen Nichols. 2012. Bufferbloat: dark buffers in the internet. *Commun. ACM* 55, 1 (2012), 57–65.
[43] Mohammad Ghasemisharif, Peter Snyder, Andrius Aucinas, and Benjamin Livshits. 2019. Speedreader: Reader mode made fast and private. In *The World Wide Web Conference*. 526–537.
[44] Google. 2019. AMP is a web component framework to easily create user-first web experiences - amp.dev. https://amp.dev. Accessed: 2019-05-05.
[45] Google. 2026. Lighthouse. https://developer.chrome.com/docs/lighthouse Accessed February 3, 2026.
[46] Aaron Grattafiori, Laurens Van Der Maaten, et al. 2024. The LLaMA 3 Herd of Models. arXiv:2407.21783 [cs.AI] https://arxiv.org/abs/2407.21783.
[47] GSMA. 2025. Barriers to Mobile Internet Adoption and Use: The State of Mobile Internet Connectivity 2025. https://www.gsmaintelligence.com/research/research-file-download?reportId=67521&assetId=67525 Accessed: 2026-02-04.
[48] GSMArena. 2025. GSMArena - Mobile Phone Specifications, News, Reviews. https://www.gsmarena.com/ Accessed: 21 Mar. 2025.
[49] Rumaisa Habib, Sarah Tanveer, Aimen Inam, Haseeb Ahmed, Ayesha Ali, Zartash Afzal Uzmi, Zafar Ayyub Qazi, and Ihsan Ayyub Qazi. 2023. A Framework for Improving Web Affordability and Inclusiveness. In *Proceedings of the ACM SIGCOMM 2023 Conference* (New York, NY, USA) *(ACM SIGCOMM '23)*. Association for Computing Machinery, New York, NY, USA, 592–607. https://doi.org/10.1145/3603269.3604872
[50] Huawei Technologies Co., Ltd. 2025. Mobile WIFI 4G Router Huawei E8372h-517 WiFi Modem. Online. https://sincereonetech.com/products/4g-router-huawei-e8372h-517-wifi-modem/ Accessed: 25 March 2025.
[51] Institut National de la Statistique du Cameroun. 2024. *ECAM 5: KEY INDICATORS*. INS Cameroon. https://ins-cameroun.cm/en/statistique/ecam-5-key-indicators/ Accessed: 2026-02-03.
[52] Soonwon Ka, Tae Hyun Kim, Jae Yeol Ha, Sun Hong Lim, Su Cheol Shin, Jun Won Choi, Chulyoung Kwak, and Sunghyun Choi. 2016. Near-ultrasound communication for tv's 2nd screen services. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. 42–54.
[53] Matt Kimball. 2025. My traceroute (MTR). https://www.bitwizard.nl/mtr/.
[54] Jesutofunmi Kupoluyi, Moumena Chaqfeh, Matteo Varvello, Russell Coke, Waleed Hashmi, Lakshmi Subramanian, and Yasir Zaki. 2022. Muzeel: assessing the impact of JavaScript dead code elimination on mobile web performance. In *Proceedings of the 22nd ACM Internet Measurement Conference*. 335–348.
[55] Tofunmi Kupoluyi, Moumena Chaqfeh, Matteo Varvello, Waleed Hashmi, Lakshmi Subramanian, and Yasir Zaki. 2021. Muzeel: A Dynamic JavaScript Analyzer for Dead Code Elimination in Today's Web. *arXiv preprint arXiv:2106.08948* (2021).
[56] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen. 2019. Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In *Proceedings of the 26th Annual Network and Distributed System Security Symposium (NDSS 2019)*. https://doi.org/10.14722/ndss.2019.23386
[57] Hyewon Lee, Tae Hyun Kim, Jun Won Choi, and Sunghyun Choi. 2015. Chirp signal-based aerial acoustic communication for smart devices. In *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 2407–2415.
[58] Ye Geoffrey Li and Gordon L Stuber. 2006. *Orthogonal frequency division multiplexing for wireless communications*. Springer Science & Business Media.
[59] LineageOS. 2024. LineageOS Android Distribution. https://lineageos.org/ A free and open-source operating system for various devices, based on the Android mobile platform.
[60] Ivano Malavolta, Kishan Nirghin, Gian Luca Scoccia, Simone Romano, Salvatore Lombardi, Giuseppe Scanniello, and Patricia Lago. 2023. JavaScript Dead Code Identification, Elimination, and Empirical Assessment. *IEEE Transactions on Software Engineering* (2023).
[61] Tata Mbunwe. 2024. *NIS reveals 38.6% of Cameroonians living below poverty line in 2022*. Mimi Mefo Info. https://mimimefoinfos.com/nis-reveals-38-6-of-cameroonians-living-below-poverty-line-in-2022/ Accessed: 2026-02-03.
[62] Bryan McQuade and Barry Pollard. 2020. How the Core Web Vitals metrics thresholds were defined. https://web.dev/articles/defining-core-web-vitals-thresholds Accessed: 2025-04-16.
[63] MTN Cameroon. 2025. MTN Cameroon Official Website. https://mtn.cm/ Accessed: 2025-03-20.
[64] Ravi Netravali, Ameesh Goyal, James Mickens, and Hari Balakrishnan. 2016. Polaris: Faster Page Loads Using Fine-grained Dependency Tracking. In *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. USENIX Association, Santa Clara, CA. https://www.usenix.org/conference/nsdi16/technical-sessions/presentation/netravali
[65] OOKLA. 2025. Speedtest® CLI, Internet connection measurement for developers. https://www.speedtest.net/apps/cli.
[66] OpenAI. 2024. OpenAI API Reference. https://platform.openai.com/docs/api-reference/chat Accessed: 2025-03-24.
[67] Derek T. Otermat, Carlos E. Otero, and Ivica Kostanic. 2015. Analysis of the FM radio spectrum for Internet of Things opportunistic access via Cognitive Radio. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*. 166–171. https://doi.org/10.1109/WF-IoT.2015.7389046
[68] Ayush Pandey, Rohail Asim, Khalid Mengal, Matteo Varvello, and Yasir Zaki. 2024. SONIC: Connect the Unconnected via FM Radio & SMS. In *Proceedings of the 20th International Conference on Emerging Networking EXperiments and Technologies* (Los Angeles, CA, USA) *(CoNEXT '24)*. Association for Computing Machinery, New York, NY, USA, 41–47. https://doi.org/10.1145/3680121.3697812
[69] Ayush Pandey, Matteo Varvello, Syed Ishtiaque Ahmed, Shurui Zhou, Lakshmi Subramanian, and Yasir Zaki. 2025. MAML: Towards a Faster Web in Developing Regions. In *Proceedings of the ACM on Web Conference 2025* (Sydney NSW, Australia) *(WWW '25)*. Association for Computing Machinery, New York, NY, USA, 727–739. https://doi.org/10.1145/3696410.3714584
[70] Press Information Bureau, Government of India. 2023. 28 Regional Doordarshan Channels to Become HD Program Production Capable. Press release, Press Information Bureau (PIB), Government of India. https://www.pib.gov.in/PressReleasePage.aspx?PRID=1889147&reg=3&lang=2 Accessed: February 02, 2026.
[71] Zhen Qin, Qingliang Zeng, Yixin Zong, and Fan Xu. 2021. Image inpainting based on deep learning: A review. *Displays* 69 (2021), 102028. https://doi.org/10.1016/j.displa.2021.102028
[72] Ahmad Rahmati, Lin Zhong, Venu Vasudevan, Jehan Wickramasuriya, and Daniel Stewart. 2010. Enabling pervasive mobile applications with the FM radio broadcast data system. In *Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications* (Annapolis, Maryland) *(HotMobile '10)*. Association for Computing Machinery, New York, NY, USA, 78–83. https://doi.org/10.1145/1734583.1734603
[73] Rick and Morty Wiki Contributors. 2025. Dimension C-137. https://rickandmorty.fandom.com/wiki/Dimension_C-137 Accessed: 24 Mar. 2025.
[74] Nirupam Roy, Haitham Hassanieh, and Romit Roy Choudhury. 2017. Backdoor: Making microphones hear inaudible sounds. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*. 2–14.

[75] Olivier Rukundo and Hanqiang Cao. 2012. Nearest Neighbor Value Interpolation. *International Journal of Advanced Computer Science and Applications* 3, 4 (2012). https://doi.org/10.14569/IJACSA.2012.030405
[76] Kimberly Ruth, Aurore Fass, Jonathan Azose, Mark Pearson, Emma Thomas, Caitlin Sadowski, and Zakir Durumeric. 2022. A world wide view of browsing the world wide web. In *Proceedings of the 22nd ACM Internet Measurement Conference* (Nice, France) *(IMC '22)*. Association for Computing Machinery, New York, NY, USA, 317–336. https://doi.org/10.1145/3517745.3561418
[77] Salamek. 2025. Huawei LTE API. GitHub repository. https://github.com/Salamek/huawei-lte-api Accessed: 25 March 2025.
[78] G Enrico Santagati and Tommaso Melodia. 2016. A software-defined ultrasonic networking framework for wearable devices. *IEEE/ACM Transactions on Networking* 25, 2 (2016), 960–973.
[79] Aaron Schulman, Dave Levin, and Neil Spring. 2014. RevCast: Fast, private certificate revocation over FM radio. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. 799–810.
[80] ScienceDirect. 2025. Frequency-shift keying. https://www.sciencedirect.com/topics/computer-science/frequency-shift-keying. Accessed: 2025-07-25.
[81] We Are Social and Meltwater. 2026. Digital 2026 Global Overview Report. https://datareportal.com/reports/digital-2026-global-overview-report Retrieved on 04 Feb 2026.
[82] Isidora Stanković. 2014. Recovery of Images with Missing Pixels using a Gradient Compressive Sensing Algorithm. *ArXiv* abs/1407.3695 (2014). https://api.semanticscholar.org/CorpusID:15997634
[83] Kyoungwon Suh, Christophe Diot, Jim Kurose, Laurent Massoulie, Christoph Neumann, Don Towsley, and Matteo Varvello. 2007. Push-to-peer video-on-demand system: Design and evaluation. *IEEE Journal on Selected Areas in Communications* 25, 9 (2007), 1706–1716.
[84] M. Suka, Y. Hirakoso, Y. Takahashi, T. Kuroda, M. Takada, M. Fujiwara, Y. Tomida, and S. Okada. 1994. Development of DARC decoding LSI for high-speed FM subcarrier system. *IEEE Transactions on Consumer Electronics* 40, 3 (1994), 570–579. https://doi.org/10.1109/30.320843
[85] Martin Thomson and Christopher A. Wood. 2024. *Oblivious HTTP*. RFC 9458. Internet Engineering Task Force (IETF). https://www.ietf.org/rfc/rfc9458.html Proposed Standard, ISSN 2070-1721.
[86] Landlocked Developing Countries United Nations Office of the High Representative for the Least Developed Countries and Small Island Developing States (OHRLLS). 2025. List of Least Developed Countries (LDCs). https://www.un.org/ohrlls/content/list-ldcs Accessed: 2025-03-19.
[87] Matteo Varvello, Hyunseok Chang, and Yasir Zaki. 2022. 'Performance characterization of videoconferencing in the wild. In *Proceedings of the 22nd ACM Internet Measurement Conference*. 261–273.
[88] Matteo Varvello and Yasir Zaki. 2022. A Worldwide Look Into Mobile Access Networks Through the Eyes of AmiGos. In *TMA 2023 - Proceedings of the 7th Network Traffic Measurement and Analysis Conference*. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.23919/TMA58422.2023.10198920
[89] Tanya Verma and Sudheesh Singanamalla. 2020. *Improving DNS Privacy with Oblivious DoH in 1.1.1.1*. https://blog.cloudflare.com/oblivious-dns/ Cloudflare blog post.
[90] Morgan Vigil, Elizabeth Belding, and Matthew Rantanen. 2016. Repurposing FM: Radio Nowhere to OSNs Everywhere. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing* (San Francisco, California, USA) *(CSCW '16)*. Association for Computing Machinery, New York, NY, USA, 1260–1272. https://doi.org/10.1145/2818048.2820001
[91] Hans Wackernagel. 1995. *Ordinary Kriging*. Springer Berlin Heidelberg, Berlin, Heidelberg, 74–81. https://doi.org/10.1007/978-3-662-03098-1_11
[92] Xiao Sophia Wang, Arvind Krishnamurthy, and David Wetherall. 2016. Speeding up Web Page Loads with Shandian. In *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. USENIX Association, Santa Clara, CA, 109–122. https://www.usenix.org/conference/nsdi16/technical-sessions/presentation/wang
[93] WorldData.info. 2025. WorldData.info: The World in Numbers. https://www.worlddata.info/ Accessed: 2025-04-08.
[94] Hang Yu, Ahmad Rahmati, Ardalan Amiri Sani, Lin Zhong, Jehan Wickramasuriya, and Venu Vasudevan. 2011. Data broadcasting using mobile FM radio: design, realization and application. In *Proceedings of the 13th International Conference on Ubiquitous Computing* (Beijing, China) *(UbiComp '11)*. Association for Computing Machinery, New York, NY, USA, 247–256. https://doi.org/10.1145/2030112.2030146

A WebP Compression

Figure 16 shows the CDF of the size of WebP images of rendered webpages, as benchmarked in [68], under different image quality settings (Q). It also compares the pixel height (PH) of the image, which is either set to "none", i.e., the maximum loaded length of the webpage, or cropped at 10K pixels. At a low quality setting of 10, most webpages can be compressed to less than 200 KB (almost 10× compression), whereas a high quality setting of 90 requires approximately 700 KB. The results further show that cropping webpages to 10K pixels reduces the image size by approximately 100 KB for 75% of webpages. The tails of the CDFs indicate that some images are about twice as large as the 90th percentile, for example 500 KB instead of 250 KB when Q is set to 10 and PH is set to 10K. Such images may require substantial delivery time, up to 6–7 minutes under a single delivery frequency and a 10 kbps rate and may therefore benefit from further optimizations.

B Pixel Interpolation

Figure 17 contains the results from a user study conducted in [68] to benchmark the effectiveness of the pixel interpolation approach. The study simulates varying levels of visual loss on popular Pakistani webpages and evaluates their impact on perceived readability and content clarity. Screenshots of the top 50 webpages from the Tranco list were captured and processed under four levels of synthetic visual loss (5%, 10%, 20%, and 50%). Each screenshot was rendered in two variants—one with missing pixels left dark, and another corrected using nearest-neighbor pixel interpolation (detailed in Section 4.2)—yielding 400 total images.

A total of 151 university students in Pakistan participated in the study, each rating 20 randomly assigned screenshots such that each image received at least seven responses. Ratings were collected on a 0–10 Likert scale across two dimensions: (a) content clarity (*How well can you understand the content in this image?*), and (b) text readability (*How readable is the text in the image given the noise?*). Figure 17 presents median boxplots per webpage, with hatched bars representing content clarity and plain bars representing text readability. Even with a 20% pixel loss rate, participants reported a median content clarity score of 7 out of 10, suggesting that the overall understanding of the page remained largely intact. Although text readability was more impacted, it continued to be within an acceptable range at 20% loss.

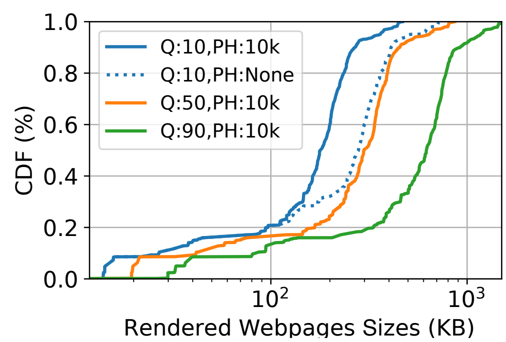


Figure 16: CDF of the size of WebP screenshots of rendered webpages assuming variable image quality (Q) and pixels height (PH).

C Comparison to SONIC

Figure 18 contrasts the full server–client architecture of WebFM with SONIC [68]. Components shown in red correspond to SONIC, while those in blue are unique to WebFM. SONIC was presented as a position paper and proof-of-concept rather than a deployable system. It did not involve a live FM broadcast, real users, or sustained transmissions over operational radio infrastructure. Instead, SONIC simulated webpage transmissions, relayed audio from an external FM radio to smartphones via an auxiliary cable, and evaluated decoding through short-range experiments and under lossless conditions. As a result, it could not capture the effects of geographically induced signal variability, which fundamentally shape reliability in real FM deployments. In contrast, WebFM is designed to operate under these constraints. By modifying LineageOS to expose built-in FM tuners on commodity smartphones, WebFM integrates at the operating system level and removes external hardware dependencies, enabling direct FM-transmitter-to-phone reception. Moreover, WebFM's file format and decoding pipeline are explicitly informed by loss patterns observed during real-world deployments, enabling partial recovery and graceful degradation under varying signal conditions.

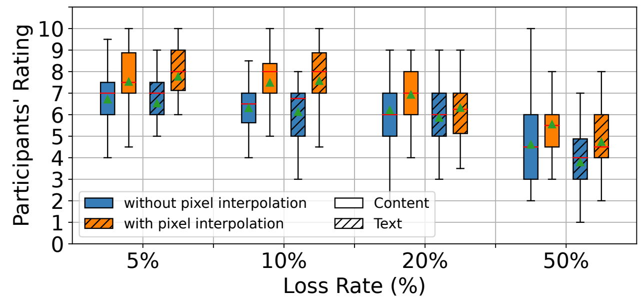


Figure 17: Distribution of user ratings (0–10) for top 50 Pakistani webpages, with/without interpolation.

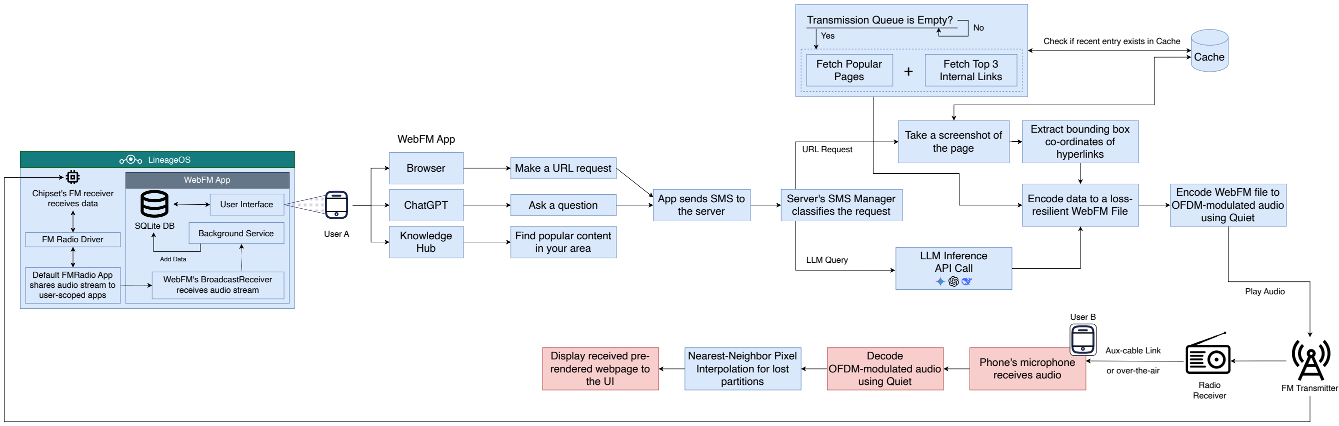


Figure 18: Comparison of WebFM and SONIC. Highlighted in red are SONIC's components; those in blue are WebFM.