DEVELOPMENT OF A WEB-BASED PLATFORM FOR CELLULAR NETWORK PERFORMANCE REVIEW USING CROWDSOURCED DATA

BY

ABDULROFIHI MUSA ALAO
IFS/19/0579

SUPERVISED BY

DR. T.E. BALOGUN

DEPARTMENT OF INFORMATION SYSTEMS,
SCHOOL OF COMPUTING,
THE FEDERAL UNIVERSITY OF TECHNOLOGY, AKURE

IN PARTIAL FULFILMENT
OF THE REQUIREMENT FOR THE AWARD OF A
BACHELOR OF TECHNOLOGY (B. TECH) DEGREE IN
INFORMATION SYSTEMS

JULY 2025

CERTIFICATION

This is to certify that this project was carried out by **ABDULROFIHI MUSA ALAO** with the Matric Number, **IFS/19/0579,** and submitted to the Department of Information Systems, School of Computing, Federal University of Technology Akure, Ondo State.

ABDULROFIHI MUSA A.	**Date**
IFS/19/0579	

DR. T.E. BALOGUN	**Date**
Supervisor	

DEDICATION

dedication….

ACKNOWLEDGEMENT

….acknowledgement

ABSTRACT

Reliable mobile internet is now essential for daily life in Nigeria, yet users often struggle to know which network truly performs best in their area. Existing coverage maps published by operators seldom reflect real-world experience, and single-shot speed tests give only a snapshot in time. This study sets out to build a simple web platform that gathers crowdsourced network measurements from users running quick tests on their phones and turns them into clear, location-based data that anyone can consult before choosing or switching providers.

The platform collects two forms of data: user-initiated speed tests (download, upload, latency and jitter), and their locations.

To be continued…

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER ONE: INTRODUCTION

1.0 Background of the Study

Reliable cellular network performance is a fundamental requirement in today's digital economy, directly influencing communication, education, business, and overall user experience. This urgency is underscored by the sheer volume of data now traversing Nigerian networks. In May 2025 alone, Nigerians consumed 1.04 million terabytes of mobile data, an all-time monthly high, followed by continued heavy usage into June (Nairametrics. 2025).

Assessing network quality, however, remains a challenge. Users typically rely on trial and error, switching between service providers based on personal experience or anecdotal evidence. While some telecom companies provide coverage maps, they do not represent user experiences, failing to reflect localized performance fluctuations caused by congestion, infrastructure limitations, or environmental factors. Ookla's "Coverage Score" methodology fuses billions of daily background coverage scans with consumer-initiated tests to expose coverage gaps in the United States, but Nigeria lacks any publicly available, continuously updated analogue of this crowdsourced, location-level dataset (Ookla. 2025). Independent speed test applications such as Fast.com offer point-in-time performance metrics but do not aggregate data at scale for location-based decision-making.

A web-based cellular network performance review platform using crowdsourced data offers a promising alternative. By collecting real-time data from multiple users, such an approach enables a dynamic, large-scale analysis of network quality across different locations. Crowdsourcing has been successfully applied in traffic monitoring (e.g., Google Maps, Waze) and air quality tracking, demonstrating its effectiveness in generating user-driven, high-resolution data models. Recent work on anticipatory vehicular communication systems by Sliwa et al. (2019) further shows that crowdsourcing-based data maintenance can keep simulations consistent with the real world and enable an up-to-date digital twin of the network. Applying this methodology to cellular network analysis can provide a more accurate, transparent, and up-to-date representation of network performance.

Federal University of Technology, Akure (FUTA) provides a suitable environment for evaluating this approach. Prior research has already used FUTA as a case study for network performance evaluation. Akintola and Eleazar (2019) analyzed the fiber-optic network on campus using simulation tools, focusing on power loss, bit-error rate, and Q-factor. Building on this foundation, the present study will leverage FUTA's diverse user base which includes students, faculty, and staff with varying network-usage patterns to assess the effectiveness of crowdsourced network-performance mapping in generating actionable insights for users.

This project aims to bridge the gap between user experience and data-driven network assessment by developing a web-based cellular-network performance-review platform that relies on crowdsourced data. Unlike traditional telecom coverage maps, this approach prioritizes real-world user data, empowering individuals with reliable, localized insights to make informed decisions about their mobile-network choices.

1.1 Motivation for Research

Existing methods for assessing cellular network performance in Nigeria are inadequate for real-world, user-centric decision-making. Among major network providers, only MTN offers a coverage map, but this seldom reflects the actual user experience at specific locations, as explained in the Introduction. Other providers do not publicly offer network coverage reports, leaving users with no official reference for connectivity quality.

Additionally, widely used speed test applications, such as Ookla and Fast.com, provide only isolated, momentary snapshots of network conditions. These tools do not account for network fluctuations over time, localized congestion, or environmental factors affecting performance. As a result, users are forced to rely on trial and error and anecdotal evidences when choosing a network provider.

This study addresses these gaps by developing a web-based cellular network performance review platform using crowdsourced data. By leveraging crowdsourced contributions, it seeks to provide a transparent, dynamic, and location-aware alternative to existing evaluation methods, ensuring that users can make informed decisions based on collective real-world experience rather than incomplete or non-existent official data.

1.2 Aims and Objectives

This study aims to develop a web-based cellular network performance review platform by leveraging user-contributed data. By focusing on FUTA as a case study, the research seeks to demonstrate the feasibility of a user-driven approach to network performance assessment, addressing the lack of transparent and localized data for informed decision-making.

To achieve this aim, the study will pursue the following objectives:

I. Identify limitations of existing network assessment methods.

II. Design and develop a web-based application for real-time network monitoring.

III. Implement a data aggregation system for meaningful insights.

IV. Evaluate the platform’s accuracy within FUTA’s network environment.

V. Compare results with traditional speed test tools.

1.3 Methodology

The methodology adopted for this project centres on developing a web-based application to monitor cellular network performance using crowdsourced speed test data, aligning with the objectives of real-time monitoring, data aggregation, platform evaluation, and comparison with traditional tools. It has the following stages:

I. Requirement and Planning: The requirements and planning stage involves the gathering and comprehensive evaluation of functions the system must perform and the needs of the target users.

II. Literature Review: This stage involves reviewing existing literature on implementing internet-speed measuring systems, and relevant software engineering principles. By examining case studies of similar projects to identify best practices and potential challenges, the synthesized findings will be used to inform the design and development stages.

III. Design: In this stage, the architecture for the system is developed, outlining its components which include a user interface, a backend and a database. The interactions between these components will also be highlighted in this stage.

IV. Implementation: This stage implements the project based on the design specifications from the previous stage. This is where the logic for all the system components comes alive.

V. Testing: This stage occurs immediately after the code implementation is done. Problems in the system are checked for by specific users who find vulnerabilities or user experience issues in the system. All reported issues will be fixed before the project is pushed to production.

VI. Deployment: In this stage, the project is pushed to production for use by all users, moving the project out of the testing phase after all issues and bugs are addressed.

This methodology ensures a scalable, user-friendly application that delivers reliable insights about network performance.

1.4 Expected Contribution to Knowledge

At the end of this research, a web-based platform for cellular network performance review platform using crowdsourced data with FUTA as a case study is expected to have been developed. By leveraging crowdsourced network speed data, the study offers several key contributions:

1. **A Public, User-Driven Network Performance Index:** Unlike existing coverage maps controlled by telecom providers, this system provides real-world user experiences rather than theoretical models. The findings contribute to a more transparent and independent evaluation of internet service quality.
2. **Bridging the Information Gap for Digital Nomads, NYSC Corps and Remote Workers:** Reliable internet is critical for professionals who work remotely. However, when relocating to a new area, there is little guidance on which ISP performs best. This research provides a location-based tool that helps users make informed decisions about network providers based on actual performance data.

CHAPTER TWO: LITERATURE REVIEW

2.0 Introduction

This chapter reviews the existing body of literature relevant to the development of a web-based cellular network performance review platform using crowdsourced data. It provides a conceptual and theoretical foundation for understanding the dynamics of cellular network quality, the tools used to measure it, and the technologies that support such measurements. This literature review draws from scholarly articles, technical documentation, white papers, and related industry projects to build a comprehensive view of the subject.

2.1 Conceptual Overview

To appreciate the problem space this project seeks to address, it is necessary to understand several core concepts in network performance measurement and mobile connectivity. These include signal strength indicators, latency metrics, and tools used for network diagnostics.

2.1.1 Network Coverage Maps

Network coverage maps are visual representations provided by telecom operators or third-party platforms to show the geographical spread of their signal availability. These maps are typically derived from the signal strength of local base stations (masts) owned by Internet Service Providers (ISPs). While useful for planning and decision-making, they do not account for real-time network load, such as slowdowns caused by high user activity during the day, and therefore often fail to represent actual user experience.

2.1.2 Latency (Ping)

Latency, often referred to as "ping," measures the time it takes for a data packet to travel from a user's device to a server and back. It is usually expressed in milliseconds (ms). Lower latency indicates a more responsive network, which is critical for activities like video calls, gaming, or live-streaming. High latency can result in noticeable lags and poor user experience, even when download speeds are acceptable.

2.1.3 Jitter

Jitter is the variation in latency over time. In a stable connection, packets reach their destination at consistent intervals. However, network congestion or instability can cause packets to arrive irregularly, which affects real-time communications. High jitter can lead to choppy audio, frozen video frames, or data retransmissions.

2.1.4 Speed Testing

Speed testing tools measure the upload and download rates of a connection, often by sending data to and from nearby servers. These tools typically report three key values: download speed, upload speed, and ping. However, results can vary depending on server location, network congestion, device performance, and even the time of day.

Understanding these core concepts is essential to framing this project, which aims to bridge the gap between generalized network availability data and the lived reality of users in specific locations. With this conceptual base, this project leverages practical metrics to guide users toward better mobile connectivity decisions.

2.2 Related Works

Engin and Ahmet (2020) analyzed the reliability and packet loss (PL) performance of Mobile Network Operators (MNOs) across Turkey using 648,966 measurements from 209,776 devices over 13 months. Using statistical methods, including confidence interval (CI) analysis, the study finds that MNO-3 consistently outperforms MNO-1 and MNO-2, with rankings remaining stable even when latency and PL thresholds are adjusted. The study highlights regional disparities in network reliability across urban, suburban, and rural areas. Key limitations include potential selection bias, as the dataset comes from users who installed a network performance test app. The study also focuses only on 4G networks, excluding 5G, and its findings may not generalize beyond Turkey. Nonetheless, the research provides a strong methodological framework for leveraging crowdsourced data in network performance evaluation.

Dahunsi and Kolawole (2015) introduced a crowdsourcing model to assess mobile network performance in Nigeria. The research addresses the disconnect between telecom infrastructure capabilities and users' actual Quality of Service (QoS) experiences. To bridge this gap, the authors developed NetworkQoS, an Android application that collects real-time, location-specific network performance data from volunteer users. Their study evaluates Key Performance Indicators (KPIs) set by the Nigerian Communications Commission (NCC), including Call Setup Success Rate (CSSR), Dropped Call Rate (DCR), Traffic Channel Congestion (TCH-CONG), and Handover Success Rate (HSR). Data from NetworkQoS, tested on three different Mobile Network Operators (MNOs), revealed significant variations in signal strength, network types, and congestion levels during peak hours. A key conclusion is that crowdsourced QoS measurement provides a more user-centric and real-time alternative to traditional methods like drive testing.

Dahunsi and Akinlabi (2019) assessed the quality of mobile broadband services in Nigeria, focusing on 2G and 3G networks. Their study aimed to address the lack of customer-centric performance data by developing a mobile broadband performance measurement application, MBPerf, to collect key Quality of Service (QoS) metrics such as download and upload speeds, latency, and DNS lookup times. Using a crowdsourced approach, the MBPerf app was installed on 100 Android smartphones in Akure and Ibadan, gathering data hourly over three months. The collected data was analyzed using Microsoft Excel and SPSS. Findings revealed that 3G users frequently experienced speeds below industry benchmarks, with significant performance fluctuations, especially during peak hours, while 2G users achieved speeds exceeding their minimum benchmark expectations. The study also found that network performance improved in the early morning hours, and DNS servers functioned efficiently without introducing major delays. Despite its insights, the study acknowledges certain limitations, including potential biases due to reliance on volunteers and the exclusion of 4G networks, which were not widely available at the time. The authors recommend strategies such as network densification, increased spectrum allocation, and improved capacity management to enhance broadband quality. They also suggest that regulatory bodies like the Nigerian Communications Commission (NCC) implement continuous performance monitoring systems to foster transparency and competition among mobile network operators.

Benedict and Prof. Omije proposed a new approach to evaluating and ranking Mobile Network Operators (MNOs) by developing a Quality of Service Key Performance Indicator (QoSKPI) Index. Their study aimed to address the limitations of existing methods that assess QoS parameters in isolation by integrating multiple key performance indicators into a single metric. Using the Analytical Hierarchy Process (AHP), the authors construct the QoSKPI Index based on Call Setup Success Rate (CSSR), Call Drop Rate (CDR), Standalone Dedicated Control Channel Congestion (SDCCH), and Traffic Channel Congestion (TCH). The index is validated through a comparison with consumer complaints and empirical data, demonstrating its effectiveness in capturing real-world network performance. The findings suggested that the QoSKPI Index provides a more accurate and comprehensive means of ranking MNOs, offering insights that can influence both regulatory strategies and consumer decision-making. Despite its promising results, the study acknowledged limitations, primarily the reliance on data from 2G networks, which may not fully represent the performance of modern mobile networks. The authors recommend further research to extend the index to cover 3G, 4G, and emerging 5G technologies. They also suggested that the Nigerian Communications Commission (NCC) adopt the QoSKPI Index as a regulatory tool and encourage MNOs to leverage it for network optimization. By implementing this index, the study argued that competition among operators could be strengthened, ultimately leading to improved service quality and a better user experience.

Marina et al. (2015) examined how indoor and outdoor environments affect crowdsourced mobile network measurements. They highlighted the increasing use of smartphones as measurement tools for mobile network analysis, which enables cost-effective and fine-grained coverage assessment. However, they argued that failing to distinguish between indoor and outdoor data can lead to misleading results. Using a dataset from OpenSignal containing nearly 8 million measurements from central London, the authors show that signal strength (RSSI) varies significantly between the two environments, with median differences exceeding 15 dBm. They demonstrated that conflating these measurements can lead to inaccurate assessments, often overestimating indoor coverage while underestimating outdoor coverage. To validate their findings, the authors conducted controlled measurements with ground truth indoor/outdoor context, confirming the necessity of differentiating between the two environments for accurate coverage analysis. They also discussed the challenges of designing energy-efficient and accurate context detection mechanisms in crowdsourcing-based systems. The study concludes that indoor-outdoor context awareness is critical for reliable mobile network performance evaluation and calls for future research into robust, context-aware measurement systems that can improve the accuracy of crowdsourced network analysis.

Caldas et al. (2023) examined internet speed disparities across OECD and G20 countries using Ookla's speed test data. Their study highlighted the urban-rural divide, showing that rural areas experience speeds nearly a third slower than national averages, while urban centers enjoy speeds 13% higher. The study employed two geographical frameworks, the Degree of Urbanisation and the OECD regional typology, to analyze regional differences, revealing that areas farther from metropolitan hubs suffer speeds up to 24% below national averages. Comparing Ookla's data with Denmark's *Tjekditnet.dk* validates the trends but also exposes variations in absolute speeds. While the study underscores the value of crowdsourced data in evaluating internet quality, it notes key limitations. The reliance on Ookla may introduce biases, such as an overrepresentation of faster connections, and the study's exclusion of 5G limits its scope. Additionally, its focus on relative rather than absolute speed variations complicates cross-country comparisons. Nonetheless, the findings provide actionable insights for policymakers addressing connectivity gaps in underserved regions.

Mikkelsen et al. (2020) investigate how crowdsourced data from consumer devices can be leveraged to create network performance maps for IoT applications. The research focused on cellular network variability, particularly its impact on Intelligent Transportation Systems (ITS), Smart Cities, and Smart Grids. By collecting signal strength and transport-layer round-trip time (RTT) data, the authors aim to enhance system response time predictions and improve decision-making for IoT deployments. A key finding of the study is that network performance maps are highly dependent on data density. In areas with sparse measurements, reliability decreases, prompting an exploration of whether incorporating neighboring measurements improves accuracy. The results indicated that the effectiveness of this approach varies by metric and environment. In urban areas, where signal strength fluctuates rapidly, the optimal range for incorporating neighboring data is 20–40 meters, while in rural areas, it extends to 100 meters. For RTT, the study finds that neighboring measurements remain beneficial over greater distances without introducing significant bias. The study concluded that while crowd-sourced network performance measurements are valuable for IoT applications, their implementation must be context-aware, adapting to the specific metric and environment. The authors suggest further research into initial cell size selection and the real-world applicability of these maps in optimizing IoT system performance.

Sharma et al. (2024) presented a method for defining stable, data-driven regions for internet latency analysis. Using Ookla's geolocated latency data from Chicago, the authors applied spatial interpolation techniques like Inverse Distance Weighting (IDW) and the SKATER regionalization algorithm to identify contiguous areas with similar latency characteristics. Their approach achieved a median Adjusted Rand Index (ARI) score of 0.59, indicating moderate-to-strong clustering stability. Key findings include the advantage of higher percentile latency aggregation, the superiority of hexagonal grids over census tracts for spatial analysis, and the low similarity between ISP-specific latency regions, underscoring the need for public ISP performance maps. The study highlighted its applicability for policymakers in prioritizing infrastructure investments but acknowledges limitations, such as potential biases from self-selected Ookla data and untested effectiveness in rural areas. Despite these constraints, it offered a refined framework for using crowdsourced data to assess regional internet performance and disparities more accurately.

Li et. al (2023) investigated the reliability of crowdsourced cellular signal data, particularly for basestation localization. The authors analyzed large-scale datasets from OpenCellID and OpenBMap, which include 419 million signal measurements covering nearly 1 million cells, to evaluate the effectiveness of crowdsourced data in predicting basestation locations. The study explores both supervised and unsupervised learning techniques to improve localization accuracy, focusing on identifying measurement features that correlate with high-quality data. Traditional basestation localization approaches using raw received signal strength (RSS) data yield inconsistent and inaccurate results due to noise, user behavior, and uneven spatial coverage. While supervised learning improves accuracy, it fails to reduce the variance in localization results significantly. To address this, the authors introduce an unsupervised learning approach using feature clustering, which reveals that RSS standard deviation and RSS-weighted dispersion mean are key indicators of high-quality measurements. By optimizing crowdsourced data selection based on these features, the study achieves a notable improvement in localization accuracy and a significant reduction in error variance. Furthermore, the authors propose an adaptive crowdsourcing technique that dynamically filters low-quality measurements, enhancing the reliability of the dataset. Their findings are validated across multiple datasets and geographic regions, demonstrating that the proposed techniques generalize well. While crowdsourced signal measurements are inherently noisy, the study concludes that careful feature selection and adaptive data filtering can unlock their potential for more accurate and reliable basestation localization.

Bischof et al. (2011) proposed a novel method for evaluating Internet Service Providers (ISPs) by leveraging network-intensive applications on end-user devices. They argued that traditional ISP characterization methods, such as web-based speed tests and dedicated hardware probes, struggle to balance coverage, continuous monitoring, and user-perceived performance. To address this, they introduce C2E (Crowdsourced ISP Characterization at the Network Edge), a system that passively analyzes user-generated traffic from applications like BitTorrent to provide scalable and real-time ISP performance insights. Using a dataset from 500,000 BitTorrent users spanning 3,150 networks, the study demonstrates that C2E effectively identifies ISP service tiers, tracks time-of-day performance fluctuations, and detects service outages from an end-user perspective. The authors emphasize that this approach offers a unique, bottom-up perspective on ISP performance, complementing existing characterization methods. While C2E provides valuable insights into network quality at scale, its reliance on BitTorrent traffic may limit applicability in regions where such usage is restricted or not representative of broader internet activity. The study concludes that C2E enhances ISP analysis by offering real-time, user-centric data that can inform broadband policy, competitive analysis, and network performance benchmarking.

Hoßfeld et. al (2020) examined the role of crowdsourcing in network and Quality of Experience (QoE) measurements. Their study categorized crowdsourced network data as *active* (speed tests generating artificial traffic) or *passive* (cellular data collected from user activities). The study highlighted key use cases, including benchmarking for competitive analysis, network planning to identify coverage gaps, and optimization for detecting issues like congestion. Compared to traditional methods like drive testing, crowdsourcing provides broader geographic coverage but lacks granular protocol-level insights. Challenges include ensuring data validity, reliability, and representativeness while addressing ethical concerns like privacy and participant consent. Their work emphasized the importance of rigorous study design, robust data cleaning, and ethical adherence to maximize the reliability and utility of crowdsourced network measurements.

Sivachandiran et al. (2021) examined how edge computing, fog computing, IoT sensors, and cloud platforms can improve data transmission efficiency in crowdsourced networks. Their work highlighted edge computing for speed and efficiency, cloud platforms for secure storage, and IoT and fog computing for bandwidth and latency management. Their study focused on large-scale public-use cases like surveillance and social media data handling while acknowledging challenges in optimizing transmission speed. They reviewed frameworks like the *Honeybee Algorithm* for merging edge, cloud, and crowdsourcing, and *Human-Driven Edge Computing* for better data handling. They found that these integrations enhance latency, speed, and performance but introduce challenges like cost and infrastructure complexity. A key limitation is the reliance on theoretical models rather than real-world implementations, making practical deployment an area for future research.

Midoglu et al. (2017) examined how different mobile network speed measurement tools, such as Ookla Speedtest, OpenSignal, RTR-Nettest, and MobiPerf, differ in methodology and results. Using MONROE-Nettest, a configurable tool that simulates various measurement approaches, the authors conduct large-scale tests on commercial mobile networks. They analyze key factors like the number of parallel TCP flows, measurement duration, and server location to determine their impact on reported speeds. They found that measurement methodologies significantly affect results, with server proximity and the number of TCP flows playing critical roles. Their study determined that using seven TCP flows for at least eight seconds captures reliable 4G network speeds. However, discrepancies among tools highlight the need for standardized measurement practices. The paper suggests future research should focus on minimizing data consumption and enhancing result processing for more accurate and consistent mobile speed evaluations.

Feamster and Livingood (2020) critiqued the shortcomings of current speed testing tools, arguing that they often misrepresent ISP performance by capturing bottlenecks within users' home networks rather than the ISP's access link. As broadband speeds increase, traditional speed tests fail to keep pace, frequently measuring limitations in Wi-Fi performance, client hardware, or test server capacity instead of the actual network service. The authors emphasized that factors such as transit and interconnect capacity, server provisioning, and cross-traffic can significantly skew results, leading to misleading conclusions about network quality. Since governmental organizations rely on these tests for broadband policy and regulation, ensuring their accuracy is critical. To address these issues, the authors proposed a shift towards more robust, standardized testing methodologies that reflect real-world network conditions. They advocated for replacing outdated tools like NDT with more advanced techniques that account for modern high-speed networks, including dedicated hardware-based measurement devices and ISP-embedded speed test applications. Instead of relying on single-path testing, they suggest measuring speeds to multiple destinations simultaneously to mitigate routing bottlenecks. Additionally, they argued that raw speed alone is an insufficient metric and should be complemented by application performance assessments, such as video streaming quality, to provide a fuller picture of user experience. By modernizing speed testing practices, the study highlights the need for greater transparency and methodological rigor to ensure that broadband infrastructure investments and regulatory decisions are based on accurate, meaningful data.

MacMillan et al. (2022) outlined key recommendations for accurately measuring internet speed using crowdsourced data. The authors highlighted challenges such as client-server latency, access medium (WiFi vs. wired), and client device hardware, which can distort speed test results and misrepresent ISP performance. Their findings are based on controlled in-lab experiments and a six-month study across 77 Chicago households using Ookla Speedtest and M-Lab’s NDT7. The key recommendations include logging client-server latency, avoiding WiFi for high-speed links, recording client hardware and software, and running paired tests. Their study underscored that speed test data is valuable but must be collected and analyzed carefully to avoid misattributing performance issues to ISPs when factors like WiFi interference or client hardware are at fault. The authors stressed contextualizing speed test data for accurate broadband policy decisions.

Sveen (2019) investigated strategies for optimizing the storage and querying of large, heterogeneous geospatial datasets. The study introduced the Heterogeneous Open Geodata Storage (HOGS) system, a command-line utility designed to automate the import of geospatial data into PostgreSQL/PostGIS. Sveen compared a conventional table-based relational storage model with a NoSQL document-store approach, evaluating their performance in terms of import speed, query efficiency, and storage utilization. The findings revealed that for datasets that can be effectively segmented into homogeneous subsets, the traditional relational model outperforms NoSQL in both speed and efficiency. The relational model exhibits faster data imports, quicker queries, and more effective storage use, reinforcing its relevance in handling large-scale geospatial data.

Basiri et al. (2019) explored the complexities of Volunteered Geographical Information (VGI), emphasizing concerns about data quality, biases, and trust. While VGI has proven valuable for numerous applications, its reliability remains a pressing issue due to variations in contributor expertise and the lack of standardized quality assessment methods. The authors discuss how quality is often measured by comparison with authoritative datasets or through weighted user contributions, yet these approaches do not always guarantee consistency. Biases such as voluntary response bias—where only certain individuals choose to contribute—and the long tail effect, in which a small subset of users generate most of the data, further challenge the integrity of crowdsourced geospatial information. Demographic imbalances, including gender and geographic representation, add another layer of complexity, reinforcing the need for more inclusive participation. Beyond quality and biases, trust and transparency play a critical role in the usability of VGI. The study highlights concerns regarding the credibility of contributors and the importance of trust models like VGTrust, which evaluate data reliability based on user history and spatial-temporal context. The authors argue that addressing these issues is essential for the widespread adoption of VGI in decision-making processes. Looking ahead, they call for research into improving contributor diversity, enhancing trust frameworks, and clarifying intellectual property rights to ensure VGI remains a dependable and democratic source of geospatial data. While crowdsourced mapping has revolutionized data collection, its long-term viability hinges on creating systems that balance openness with rigorous quality controls.

CHAPTER THREE: SYSTEM DESIGN

3.0 Introduction

The design and implementation of a Web-based Platform For Cellular Network Performance Review Using Crowdsourced Data requires a well-structured and meticulous system design to ensure it is reliable and functional. In this section, the system design employed for this project is presented, which includes its architecture, communication and data flow, architectural considerations and database design.

3.1 System Architecture

The architecture of the Web-based Platform for Cellular Network Performance Review Using Crowdsourced Data is designed to ensure reliability, scalability, and maintainability. It follows a layered, modular design based on the client-server model, which separates concerns and allows for independent development and scaling of each component. By leveraging open-source tools and web technologies, the system provides a robust infrastructure for real-time data collection, processing, and visualization of network speeds.

The architecture is composed of three primary layers:

- **Client Layer:** The front-facing interface that users interact with.
- **Server Layer:** The backend logic that handles processing, routing, and coordination.
- **Data Layer:** This is responsible for persistent storage and retrieval of crowdsourced data.

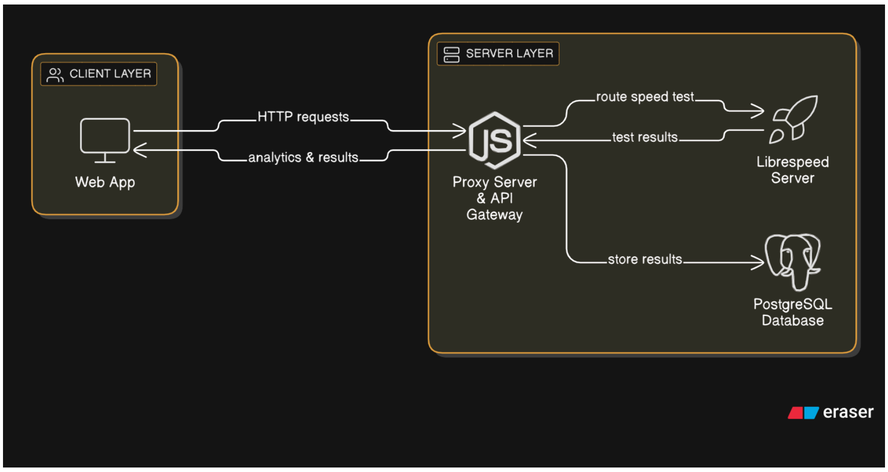

Figure 3.1: System Architecture Diagram for the Web-based Platform For Cellular Network Performance Review Using Crowdsourced Data

Each component is designed to work asynchronously and communicate over RESTful APIs, ensuring a responsive user experience while accommodating a growing volume of users and data.

3.1.1 Client Layer

The client layer consists of a web application built with modern web technologies. It serves as the entry point for user interaction, allowing users to:

- Initiate network speed tests.
- Submit test results automatically along with geolocation and device metadata.
- Review network performance data.

The application emphasizes usability, offering a lightweight interface that performs well on both desktop and mobile devices. It interacts with the backend through HTTPS requests, fetching aggregated metrics and sending user-generated data securely.

3.1.2 Server Layer

The server layer orchestrates the core operations of the system. It consists of three major components:

a. Proxy Server and API Gateway: This component is built using Express.js and acts as the central point of coordination. It performs two essential functions:

- **Proxying Speed Test Requests:** It routes user-initiated speed tests to the Librespeed server. This setup abstracts the testing logic and allows for centralized control, logging, and security enforcement.

- **Serving Analytical APIs:** The server also exposes custom API endpoints that compute key metrics such as average speeds, latency trends, and user distribution by location. These APIs power the visual analytics seen on the frontend.

By combining proxy and API capabilities in one layer, the system simplifies traffic flow and reduces latency while maintaining separation of concerns.

b. Librespeed: Librespeed is an open-source engine responsible for conducting speed tests. It performs bandwidth and latency measurements and sends raw test results to the API Gateway. Librespeed is integrated into the architecture in a headless fashion, meaning users never directly interact with it, which enhances modularity and security. This design choice enables fine-grained control, keeping the system open to future enhancements like test parameter tuning or load balancing.

c. Database: The database serves as the system's persistent data store. It captures structured records of:

- **ISP info:** This refers to the name of the Internet Service Provider (e.g., MTN, Glo, Airtel) that the user is connected to during the test. It helps identify which network is delivering the observed performance in a given location.

- **Download and upload speeds:** Download speed measures how fast data is received from the internet, while upload speed measures how fast data is sent. These values indicate the overall throughput and responsiveness of the network.
- **Ping and jitter measurements:** Ping shows the time it takes for a signal to travel from the user's device to a server and back, reflecting network latency. Jitter measures the variability in ping over time, which affects the stability of real-time activities like video calls.
- **Timestamp and approximate geolocation:** The timestamp records when a speed test was conducted, while geolocation captures the user's approximate location. Together, they help track performance trends over time and across different areas.

This data forms the foundation for the insights provided by the platform for network performance review. It is indexed and optimized for analytical queries, enabling fast generation of trends and summaries. The data model is also extensible, allowing for new metrics to be added with minimal schema changes.

Together, these components form a cohesive architecture that supports real-time speed testing and scalable data analysis. The system is intentionally designed with loose coupling between layers, making it adaptable for future growth, such as mobile app integration, machine learning-based predictions, or integration with public datasets from ISPs and regulators.

3.1.3 Communication and Data Flow

Communication within the system follows a request-response model built on RESTful HTTP. When a user initiates a speed test from the web app, the request is first sent to the Proxy Server and API Gateway, which then forwards it to the Librespeed engine. Librespeed conducts the test and returns the raw results (download speed, upload speed, ping, jitter) to the proxy, which processes the data, sends it to the frontend for display, stores it in the database along with relevant metadata such as timestamp, geolocation, and ISP info.

In parallel, the client also sends user requests to fetch aggregated performance data, such as average speeds or best-performing ISPs in a location, which are served by the analytical endpoints exposed by the proxy server. This structure ensures efficient, decoupled communication between components, with the database serving as the central point of truth. All communication is secured via HTTPS, and data is sanitized and validated at each stage to maintain integrity.

3.1.4 Architectural Considerations

The architecture of the platform is designed with scalability and extensibility in mind. Components are loosely coupled, which allows for independent scaling, maintenance, and replacement without affecting the rest of the system. For instance, Librespeed can be swapped for another test engine, or the database can be upgraded to a more powerful analytics store, without rewriting the entire codebase.

Moreover, the use of a proxy server provides an additional layer of control, security, and abstraction. It ensures that test requests can be logged, rate-limited, or filtered before reaching the core testing engine. Finally, the system supports horizontal scaling: additional proxy/API instances and database replicas can be added to handle increased load, making the architecture suitable for long-term growth and real-world deployment.

3.2 Database Design

The database design for the Web-based Platform for Cellular Network Performance Review Using Crowdsourced Data plays a vital role in ensuring structured data collection, efficient querying, and reliable long-term storage. The design emphasizes simplicity and extensibility, with the core focus on capturing relevant metrics from user speed tests along with necessary metadata. This allows for flexible querying and location-specific performance analysis across different ISPs and timeframes.

The database used in this project consists of a single, well-structured table: *speedtest_records*. This table stores detailed records of each test, including network performance metrics, ISP info, and geolocation coordinates. All records are time-stamped and indexed for efficient aggregation and retrieval.

The schema is shown below:

Field	**Data Type**	**NULL**	**Description**	**Action**	**Extra**
id	SERIAL	No	Unique record identifier	Primary Key	Auto increment
timestamp	TIMESTAMP	No	Time the speed was submitted	Default now()	
ip	TEXT	No	IP address of the user’s device		
ispinfo	TEXT	No	ISP used for the test		
latitude	DECIMAL (10, 8)	No	Approximate latitude of the test location		
longitude	DECIMAL (11, 8)	No	Approximate longitude of the test location		
ua	TEXT	Yes	User agent string (used to infer device type)		

lang	TEXT	Yes	Language setting of the user’s device/browser		
dl	TEXT	No	Measured download speed (in Mbps)		
ul	TEXT	No	Measured upload speed (in Mbps)		
ping	TEXT	No	Measured latency in milliseconds		
jitter	TEXT	No	Variance in latency measurements		
log	TEXT	Yes	Raw test logs for debugging or audit purposes		

Table 3.1: Schema for the *speedtest_records* Table

To improve performance for time-based and ISP-based queries, the table includes indexes on the timestamp and ispinfo fields. These indexes allow fast filtering of results by test time or ISP, which is especially useful for generating historical trends and handling users’ queries.

3.3 Use Case Diagram

The use case diagram models the core interactions between users and the Web-based Platform for Cellular Network Performance Review. It provides a high-level overview of how different types of users (actors) engage with the system to achieve specific goals (use cases). The primary actor in the system is end-users who perform speed tests and review network performance data. The system administrator who manages data and application performance is a secondary actor. This diagram is essential for understanding the functional scope of the application from a user-centric perspective.

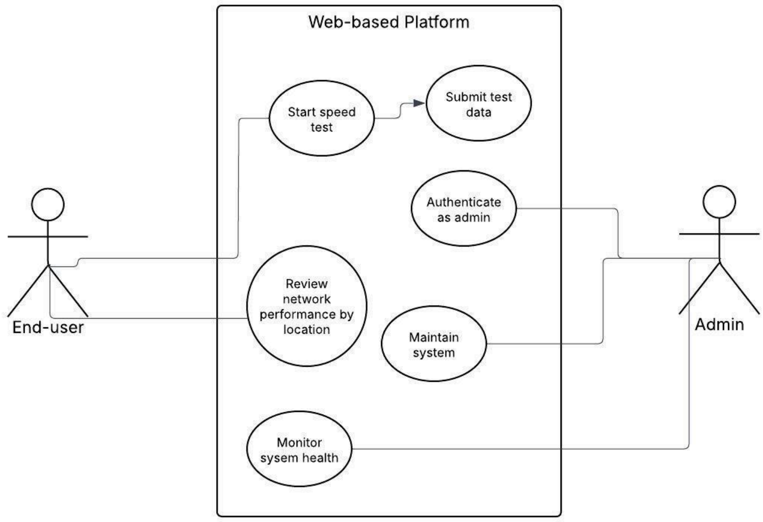

Figure 3.2: Use Case Diagram for the *Web-based Platform for Cellular Network Performance Review Using Crowdsourced Data*

a. Actors

i. Visitor: A public user of the web platform. They can test their network and review historical network performance data.

ii. Admin: A privileged user responsible for maintaining the platform, managing collected test data, and overseeing the system's operational health.

b. Use Cases

i. Submit Speed Test: Users initiate a network speed test that measures download/upload speed, ping, and jitter.

ii. View Speed Test Result: After completing a test, users receive a detailed summary of their network performance.

iii. Review Network Performance: Users can compare ISPs based on aggregated speed test data.

iv. Manage Test data: Admins can view, verify, and clean the dataset for integrity and consistency.

v. Monitor Server Performance: Admins can track backend performance, traffic loads, and resource usage.

c. Relationships

i. Association: Visitors are associated with the core frontend functionalities, submitting tests, viewing results, and exploring data.

ii. Include: Admin use cases may include background operations like logging or metrics collection

CHAPTER FOUR: SYSTEM IMPLEMENTATION

4.0 Introduction

This chapter focuses on the implementation phase of the web-based platform for cellular network performance review using crowdsourced data, where abstract design principles are translated into a working software system. It outlines the tools, technologies, and environments used to build and deploy the complete application. The implementation process leverages a monorepo structure, enabling shared configuration and seamless collaboration across services.

4.1 Development Tools

Various modern web technologies and supporting tools were employed across the frontend, backend, and deployment pipeline to develop this project. These tools were chosen for their robustness, scalability, and support for rapid development.

4.1.1 Frontend Technologies

The frontend of the application was built with core web technologies: HTML, CSS, and JavaScript. These technologies work together to provide a responsive, interactive interface through which users can run speed tests and view results. Webpack is used to bundle the frontend assets, enabling the seamless integration of Librespeed's Web Workers, a critical requirement for running tests in the browser without freezing the UI. Below is a breakdown of the frontend technologies used

- **HTML and CSS**

  HTML (HyperText Markup Language) provides the structural skeleton of web applications, defining the hierarchy and semantics of content displayed in the browser. CSS (Cascading Style Sheets) is responsible for visual styling, dictating layout, color schemes, typography, and responsive behavior across screen sizes.

  In this project, HTML and CSS were used to scaffold the user interface and implement a clean, accessible design for the network speed testing dashboard. Semantic HTML tags ensured compatibility with assistive technologies, while modular CSS rules enabled consistent styling across components. Media queries were employed to deliver a responsive experience on both desktop and mobile devices, aligning with the project's goal of wide accessibility and performance clarity.

- **JavaScript**

  JavaScript is a dynamic, high-level scripting language that enables interactive behavior in web applications. It powers DOM manipulation, asynchronous data fetching, event handling, and dynamic UI rendering directly within the browser environment.

  In this project, JavaScript was the engine behind the user interaction layer. It controlled the initiation and visualization of speed tests, managed asynchronous communication with the backend API, and handled real-time DOM updates based on test progress and results. Custom scripts were also used to collect lightweight device information for session fingerprinting, aiding in performance correlation without compromising user privacy.

- **Webpack**
  Webpack is a powerful module bundler for JavaScript applications. It compiles, bundles, and optimizes project assets—such as scripts, stylesheets, and images—into efficient, production-ready output. Its plugin ecosystem and loader system allow for deep customization of the build pipeline.

  In this project, Webpack was employed to streamline the frontend development workflow. It bundled JavaScript modules and transpiled modern syntax for cross-browser compatibility. Additionally, it processed and optimized CSS, inlined critical assets, and minimized overall bundle size to ensure a fast initial load time. Webpack's hot module replacement and source map generation also accelerated development and debugging processes, contributing to a seamless build-test-deploy cycle.

4.1.2 Backend Technologies

The backend layer of the application serves as the operational core that powers test orchestration, data persistence, and performance analysis. It comprises three key components: a custom proxy API server built with Node.js and Express.js, a PostgreSQL database for structured data storage, and a containerized instance of Librespeed for real-time network testing. These technologies work in tandem to manage user sessions, conduct and track speed tests, and provide structured performance insights via the frontend interface. This layered design ensures loose coupling between components, while enabling full control over data collection, transformation, and response formatting.

- **Node.js / Express.js (Proxy and API Server)**
  Node.js is a non-blocking, event-driven runtime environment built on the V8 JavaScript engine. It excels in handling concurrent I/O operations, making it ideal for real-time applications and network-heavy tasks. Express.js is a minimalist and flexible web framework built on top of Node.js that simplifies the creation of RESTful APIs and backend services.

  In this project, Node.js and Express.js powered the custom API server and also acted as a proxy between the frontend and the Librespeed test engine. It hosts APIs used for network performance review. The use of Express.js streamlined endpoint definitions and routing logic, allowing fine-grained control over test parameters and secure user data logging.

- **PostgreSQL**
  PostgreSQL is a powerful, open-source relational database system known for its robustness, scalability, and compliance with SQL standards. It supports advanced data types, full-text search, and JSON storage, making it suitable for a wide variety of use cases ranging from transactional systems to analytical applications.

  In this project, PostgreSQL served as the backend database for storing and querying all network test results and metadata. It enabled efficient persistence of user sessions, device fingerprints, geolocation data, and performance metrics. Its structured schema and indexing capabilities supported advanced querying and aggregation needed for analyzing network performance across different locations and timeframes.

- **Librespeed**
  Librespeed is a lightweight, open-source, self-hosted speed test utility that measures key internet performance indicators, including download speed, upload speed, ping, and jitter. It is built with simplicity and privacy in mind, allowing for full ownership of data without dependence on third-party services. Within the project, Librespeed was containerized and deployed as a backend service. The Express server proxied user-initiated test sessions to Librespeed and captured the raw output for processing and storage. This architecture allowed localized hosting of the test engine.

4.1.3 Deployment Tools

Deployment of the application was handled with tools that simplify building, containerizing, and hosting full-stack web applications. These tools helped package the frontend and backend into containerized units and deliver them efficiently to a production environment.

- **Dockerfile**
  Dockerfiles were used to create reproducible builds of both the frontend and backend components. Each service had its own Dockerfile, specifying instructions for building and running the application in a lightweight container. This ensured consistency across development and deployment environments.

  Docker also simplified local development by allowing all services to be orchestrated via docker-compose. For deployment, the same containers could be shipped to any platform supporting Docker, making infrastructure requirements less of a burden.

- **Railway**
  Railway served as the cloud platform for hosting the project. It offers a developer-friendly CI/CD workflow and support for Docker-based deployments. Railway allowed for automatic deployment of both the frontend and backend containers via GitHub integration, which was especially useful for testing and preview environments.

Logs, environment variables, and container health checks could all be monitored via Railway's dashboard, making it a convenient choice for full-stack web applications. Its support for monorepo structures and container orchestration helped maintain simplicity throughout the deployment process.

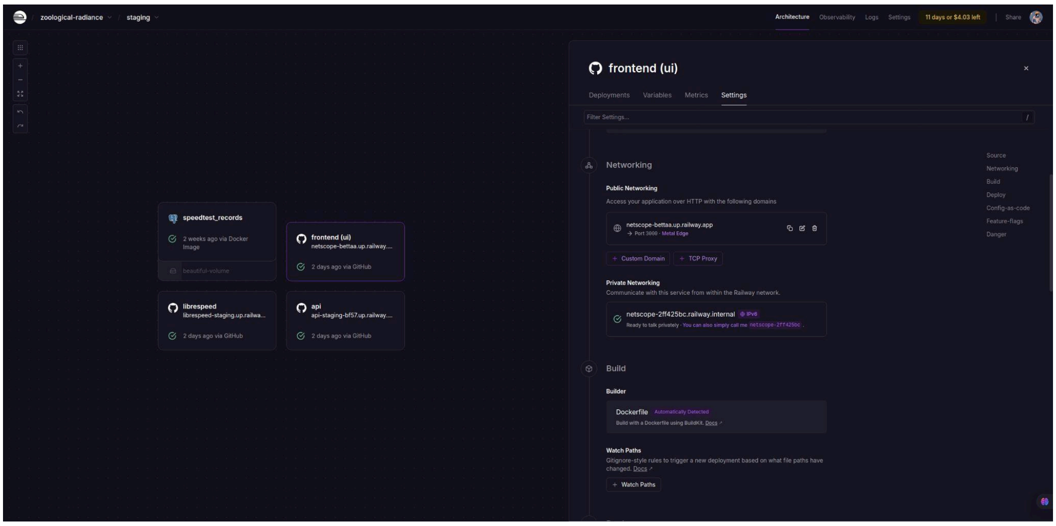

Figure 4.1: Railway Dashboard for the Project

- **Github**
  Github was the source code management tool for this project. It hosted the entire monorepo and facilitated collaboration, version control, and integration with Railway. The Github Actions feature could be optionally used for CI tasks like linting, testing, or automatic deployment triggers. Github ensured structured development practices and served as a historical log of changes made to the application.

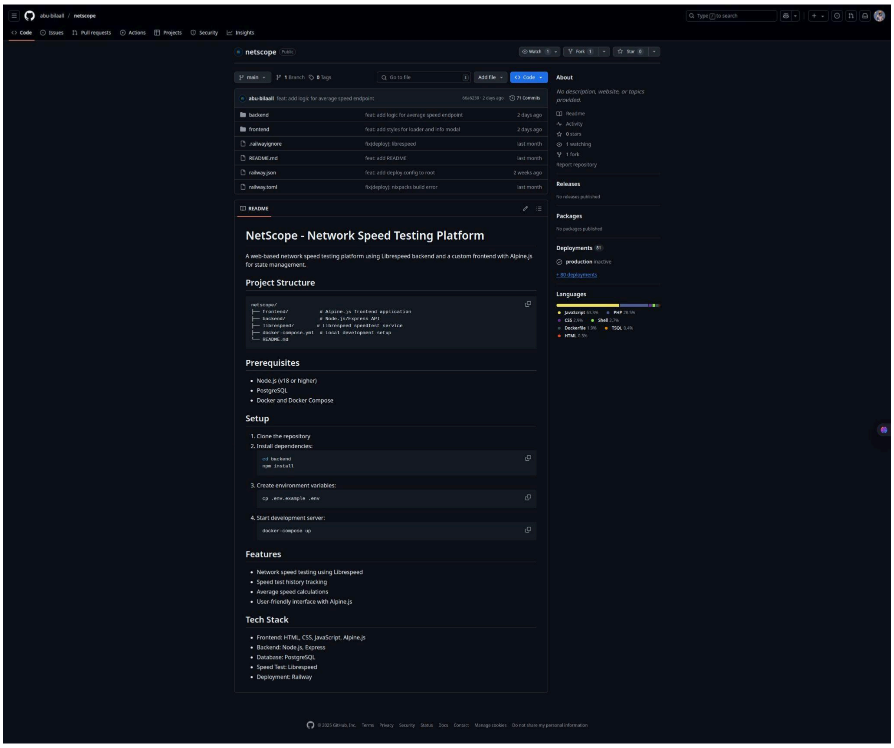

Figure 4.2: Project repo

4.1.4 Monorepo Architecture

The project used a monorepo architecture, meaning that both frontend and backend code lived in a single repository. This approach simplifies dependency management, ensures consistent versioning, and improves the developer experience by centralizing configuration and reducing boilerplate setup across multiple services.

The monorepo housed subdirectories for the frontend, proxy server, and Librespeed components, each with their own Dockerfiles and configuration files. Shared environment variables and deployment scripts could be managed centrally, reducing redundancy. This structure also worked seamlessly with Railway, which allowed for multiple services to be deployed from different subdirectories in the same repo.

4.2 System Modules

The system for the web-based platform for cellular network performance review using crowdsourced data is composed of modular, loosely coupled components, with each dedicated to a specific layer of the application. This design improves maintainability, scalability, and clarity of responsibility across the system. Here are the main modules of the system:

1. **Frontend Module**

   The frontend module is responsible for rendering the user interface and handling all user interactions in the browser. It leverages modular JavaScript, CSS, and templating to deliver a responsive, interactive experience.

   Key Components:

   - ***`src/index.js, src/js/`***: Core UI logic and event handling.
   - **`src/css/`**: Styling and layout definitions.
   - **`template.html`**: Base HTML structure rendered in the browser.
   - **`assets/`**: Icons and logos (e.g., MTN, GLO, Airtel) used for telco branding.
   - **`speedtest.js, speedtest_worker.js`**: Librespeed scripts for measuring speed in the browser.
   - **`webpack.*.js`:** Webpack configurations for bundling and optimization.
   - **`dist/`**: Production-ready build output.

2. **API Server Module**

   The API server module is a Node.js/Express backend that functions as the orchestration layer between the frontend client and Librespeed. Beyond just exposing RESTful endpoints, the server also doubles as a proxy, securely routing requests to Librespeed while handling client metadata and enforcing system logic.

   Key Components:

   - **`api/server.js`**: Main entry point for the Express application. It sets up middleware, routes, and launches the server.

- `api/routes/`: It houses the route definitions for all endpoints exposed to the frontend.
- `api/controllers/`: It contains the business logic behind each route, handling request processing and response formatting.
- `api/models/`: It holds the data schemas and structures used throughout the app.
- `api/middlewares/`: Middleware functions for request validation, logging, and error handling, ensuring reliability and security.
- `api/utils/`: Utility functions used across the backend.

4.3 Results and Discussion

This section presents the key outcomes of the system's implementation.

Location Permission Modal

This modal appears for first-time users or those who previously denied location access. It informs users that their location is used to visualize average internet speeds in their area. The modal offers two options:

- "Allow": triggers the browser's native location permission prompt.
- "Maybe Later": dismisses the modal without granting access.

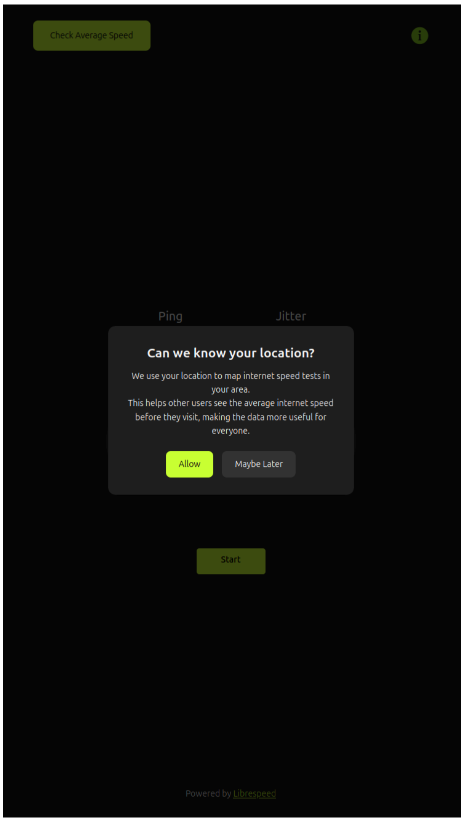

Figure 4.3: Location Permission Modal

Speed Test Page

This is the main interface for running a speed test. It features prominent gauges for Download and Upload speeds (in Mbps) and smaller displays for Ping and Jitter (in ms). A central "Start" button initiates the test, changing to an "Abort" button during the test. The name of the Internet Service Provider (ISP) used by the user for the speed test is displayed. An information icon in the header opens a modal explaining the test metrics. A "Check Average Speed" button navigates the user to the Location Speed Check page.

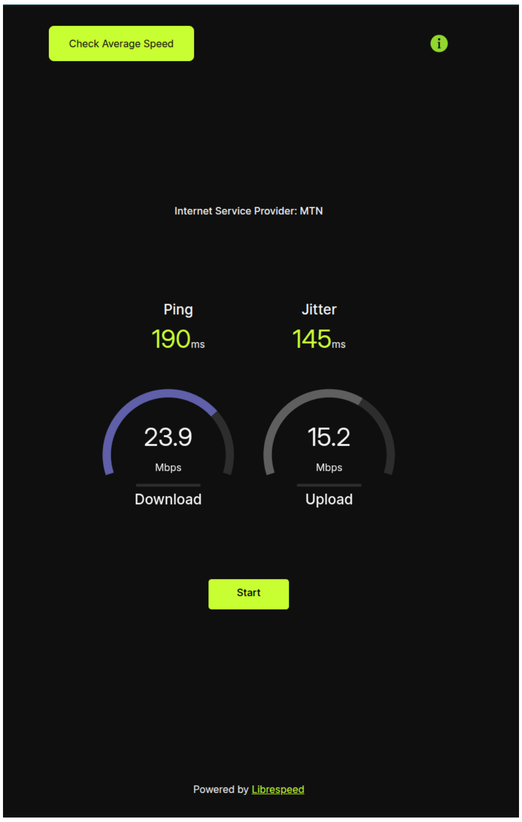

Figure 4.4: Speed test page

Location Speed Check Page

This page enables users to view average internet performance by area. A search field with Google Places Autocomplete allows users to select a location. Upon clicking "Check," the system queries the backend and displays results for major Nigerian ISPs (MTN, Glo, Airtel, 9mobile), including:

- ISP logo and name
- Average download and upload speeds in that location

If data for a particular ISP is unavailable, it is clearly indicated. The "Start Test" button returns the user to the main Speed Test Page.

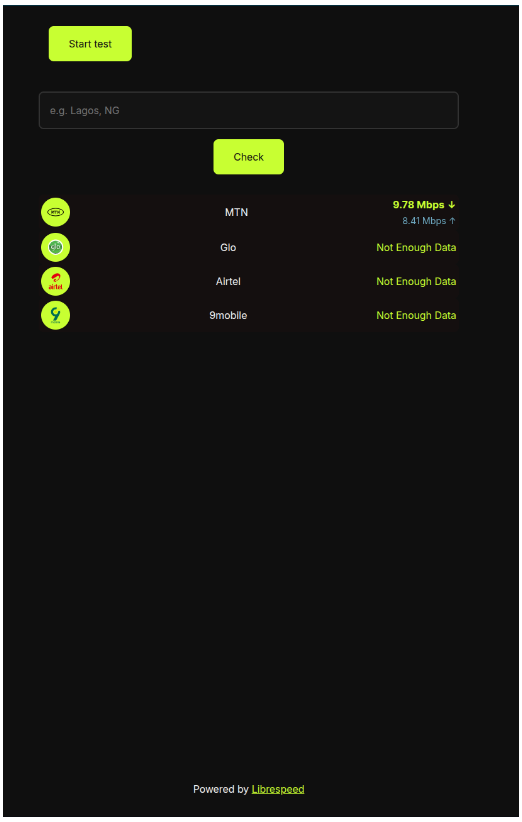

Figure 4.5: Location speed check page

Speed Test Information Modal

This informational pop-up provides users with clear, concise definitions of the key metrics measured during the speed test. It explains Jitter, Ping, Download speed, and Upload speed. The "Got it!" button closes the modal.

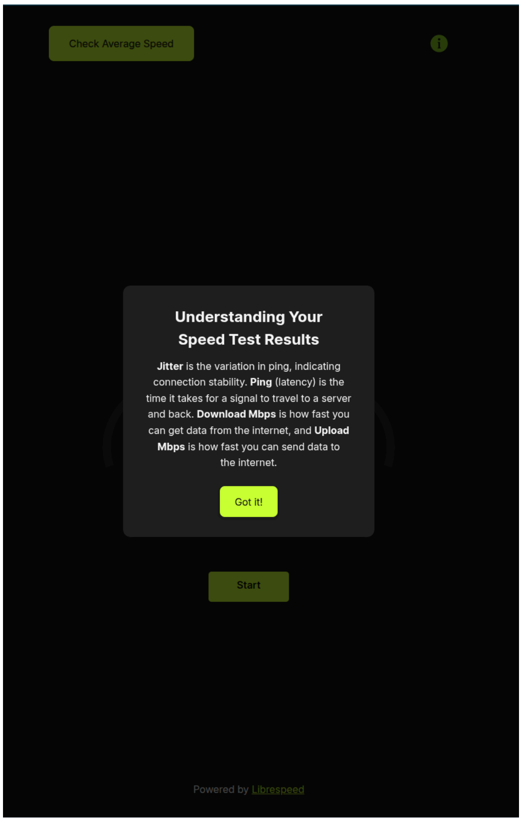

Figure 4.5: Speed test information modal

CHAPTER FIVE: CONCLUSION

To be continued…

REFERENCES

Akintola, K. G., & Eleazar, E. C. (2019). Design and performance analysis of fiber optic network system using FUTA as case study. Iconic Research and Engineering Journals, 3(5), 30–35. https://www.irejournals.com/formatedpaper/1701735.pdf

Alamri, S. (2024). The Geospatial Crowd: Emerging Trends and Challenges in Crowdsourced Spatial Analytics. ISPRS International Journal of Geo-Information, 13(6), 168.

Basiri, A., Haklay, M., Foody, G., & Mooney, P. (2019). Crowdsourced geospatial data quality: Challenges and future directions. International Journal of Geographical Information Science, 33(8), 1588-1593. https://doi.org/10.1080/13658816.2019.1593422

Benedict, O. E., & Omije, B. (2024). Improvement of mobile network performance ranking using QoSKPI Index in Nigeria. International Research Journal of Innovations in Engineering and Technology, 8(6), 130–136. https://doi.org/10.47001/IRJIET/2024.806016

Bischof, Z. S., Otto, J. S., Sánchez, M. A., Rula, J. P., Choffnes, D. R., & Bustamante, F. E. (2011). Crowdsourcing ISP characterization to the network edge. In Proceedings of the 2011 ACM SIGCOMM Workshop on Measurements Up the Stack (pp. 61–66). ACM. https://doi.org/10.1145/2018536.2018547

Caldas, M. P., Marshallan, M., & Veneri, P. (2023). Assessing spatial disparities in internet quality using speed tests. OECD Regional Development Papers, No. 47. https://dx.doi.org/10.1787/77c42f5e-en

Dahunsi, F. M., & Akinlabi, A. A. (2019). Measuring mobile broadband performance in Nigeria: 2G and 3G. Nigerian Journal of Technology, 38(2), 422–436. https://doi.org/10.4314/njt.v38i2.19

Dahunsi, F. M., & Kolawole, G. (2015). Participatory analysis of cellular network quality of service. International Journal of Computing and ICT Research, 9(1), 25–40. http://ijcir.mak.ac.ug/volume9-issue1/article3.pdf

Feamster, N., & Livingood, J. (2020). Measuring Internet speed: Current challenges and future recommendations. Communications of the ACM, 63(12), 72-79. https://doi.org/10.1145/3372135

Hoßfeld, T., Wunderer, S., & et al. (2020). Crowdsourced Network and QoE Measurements – Definitions, Use Cases and Challenges. Würzburg, Germany. doi: 10.25972/OPUS-20232

Li, Z., Nika, A., Zhang, X., Zhu, Y., Yao, Y., Zhao, B. Y., & Zheng, H. (n.d.). Identifying Value in Crowdsourced Wireless Signal Measurements. Department of Computer Science, UC Santa Barbara.

MacMillan, K., Mangla, T., Richardson, M., & Feamster, N. (2022). Best Practices for Collecting Speed Test Data. SSRN Electronic Journal.

Marina, M. K., Radu, V., & Balampekos, K. (2015). Impact of indoor-outdoor context on crowdsourcing based mobile coverage analysis. In Proceedings of the 2015 Workshop on All Things Cellular: Operations, Applications, & Challenges (pp. 31–36). ACM. https://doi.org/10.1145/2785971.2785976

Mikkelsen, L. M., Madsen, T. K., & Schwefel, H.-P. (2020). On the benefits and challenges of crowd-sourced network performance measurements for IoT scenarios. Wireless Personal Communications, 110(3), 1551–1566. https://doi.org/10.1007/s11277-019-06801-4

Midoglu, C., Wimmer, L., Lutu, A., Alay, Ö., & Griwodz, C. (2017). A Closer Look into Mobile Network Speed Measurements. ArXiv, abs/1710.07805.

Ookla. (2025, June 17). Speedtest® Methodology. https://www.ookla.com/resources/guides/speedtest-methodology

Paul, U., Liu, J., Gu, M., Gupta, A., & Belding, E. (2022). The importance of contextualization of crowdsourced active speed test measurements. In Proceedings of the 22nd ACM Internet Measurement Conference (pp. 1-16). ACM. https://doi.org/10.1145/3517745.3561441

Sharma, T., Schmitt, P., Bronzino, F., Feamster, N., & Marwell, N. P. (2024). Beyond Data Points: Regionalizing Crowdsourced Latency Measurements. Proceedings of the ACM on Measurement and Analysis of Computing Systems, 8(1), 1-26. https://doi.org/10.1145/3639040

Sivachandiran, S., Jagan, M., & Mohammed, N. (2021). Conceptual Analytics on Integration of Network Technologies with Crowdsourcing Infrastructure. In Proceedings of the 3rd International Conference on Integrated Intelligent Computing Communication & Security (ICIIC 2021).

Sveen, A. F. (2019). Efficient storage of heterogeneous geospatial data in spatial databases. Journal of Big Data, 6(1), 1-14.

Zeydan, E., & Yildirim, A. (2020). Reliability comparisons of mobile network operators: An experimental case study from a crowdsourced dataset. Turkish Journal of Electrical Engineering and Computer Sciences, 28(1), 380–393. https://doi.org/10.3906/elk-1904-143