Offline and Cross-Platform Location-based services for Healthcare: A Progressive Web Application (PWA) Approach

Soujanya Pisipati[1,2], Kapil Oberai[1], Rob Lemmens[2], Carmen Anthonj[2]

[1]Indian Institute of Remote Sensing, Dehradun, India (pisipati.soujanya45@gmail.com; kapil@iirs.gov.in)
[2]Faculty of Geo-Information Science and Earth Observation, University of Twente, Netherlands (r.l.g.lemmens@utwente.nl; c.anthonj@utwente.nl)

Keywords: Progressive Web Application, Accessibility, Location-based Services, Conceptual Framework, DoonHealth

Abstract

The digital and technological transformation around the world is shaping our daily life through smart phones and mobile applications. The introduction of wireless communication has encouraged the development of various mobile applications including automation, Global Positioning System (GPS) and Location-based services (LBS). These applications are dependent on network connectivity and the multiplicity of various device operating systems. The study explores the potential of Progressive Web Applications (PWA) to address connectivity issues and offer seamless access for users in offline scenarios across diverse devices and operating systems. The main objective of the study is to propose a conceptual framework for developing an offline and cross platform location-based services mobile application using the PWA approach. As a Proof-of-Concept, to demonstrate the practical application of the developed conceptual framework, mobile application named as "DoonHealth" is developed to provide users accessibility to the healthcare facilities in Dehradun city. The application is validated by Google lighthouse tool and has received positive feedback through a usability test with 14 participants. The results confirm that "DoonHealth" is effective in offline conditions and supports multiple device operating systems while meeting the needs of the users.

1. Introduction

In today's world, rising technology demand has encouraged the development of many mobile applications, most of which are dependent on the device operating system and network support for many functionalities. They have thus limited usage in case of no or intermittent network connectivity (Dolan, 2012). According to the International Telecommunication Union (ITU), around 37 per cent or 2.6 billion people are without a stable internet connection due to obstacles such as high cost of infrastructure, affordability of devices and regulatory barriers in network deployment (ITU, 2023). There is a need to transform the way applications are designed and developed to meet user requirements (Charkaoui et al., 2015).

Mobile applications have transformed health services, encouraging the movement of healthcare accessibility from episodic to continuous care over the years (Kratzke & Cox, 2012). Depending on only one operating system to provide the services can limit the usage of the application (Alexander et al., 2021; Das & Alam, 2014). A cross-platform application has the ability to run on various operating systems such as iOS, Android and Microsoft Windows (Mole, 2020). Location-Based Services are most widely used geospatial application that provide services based on user's current location information using on-board Global Positioning System (GPS) available in mobile devices. LBS has a wide range of applications such as healthcare, transportation, emergency and disaster management and requires an internet connection to function efficiently (Steiniger et al., 2006). Even in some major cities, the network quality varies from user to user based on their commercial carrier and geographic location. Hence, a health care delivery application built to use on various devices without the need for continuous coverage of internet can aid many users (Patrick et al., 2008).

The main objective of the study is to propose a conceptual framework towards developing an offline and cross-platform location-based services mobile application using the Progressive Web Application (PWA) approach, taking healthcare as the use case. A conceptual framework is an abstract representation to guide the development and design of a system (Jabareen, 2009). As a proof-of concept, the framework is implemented to develop "DoonHealth", a PWA to provide the users accessibility to the nearby healthcare facilities in the Dehradun city. The potential users for the application are identified to be individuals trying to search and identify the required healthcare facility in the city. The application can be utilised by both non-emergency and emergency users in a new or unfamiliar neighbourhood with limited network connectivity.

1.1 Progressive Web Application

A Progressive Web Application combines the ease of a web page (HTML, CSS, and JavaScript) with the features of a platform-specific native App, its architecture described in Figure 1. The term *progressive* refers to the application to not be bound to a single device operating system. It gives any device with a browser, the ability to access the App as a Uniform Resource Locator (URL) (Frankston, 2018). The four major components of a PWA are: App Shell, Service worker, Manifest and APIs. The *App Shell* helps construct the web page. The *Service Worker* provides the support for the application to function in offline conditions and execute the file in a separate thread than the web browser (MDN Contributors, 2023). *Manifest file* gives the ability to enable the user to install the application through a web install banner (Rêgo et al., 2019). *Application Programming Interface (APIs)* plays an essential role by updating the application cache periodically and work efficiently offline by enabling the storage and retrieval of content to provide responses.

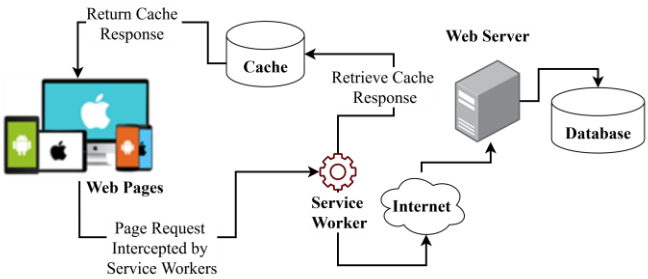

Figure 1 The Architecture of PWA development

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-5-W2-2025-521-2025 | © Author(s) 2025. CC BY 4.0 License.

Adapted from "Dawning of Progressive Web Applications (PWA): Edging Out the Pitfalls of Traditional Mobile Development" by O. Adetunji, C. Ajaegbu, N. Otuneme, O. J. Omotosho, 2020, American Scientific Research Journal for Engineering, Technology, and Sciences, 68 (1), pg-93.

Developers choose different development strategies to develop mobile application, namely, **Native App, Hybrid App, Mobile Web App and the Progressive Web Application**. Table 1 demonstrates the comparison of various features among the various strategies summarising from Oluwatofunmi Adetunji et al., (2020) and Mole, (2020).

Criteria	Native	Mobile Web App	Hybrid	PWA
Develop-ment	Using programm-ing languages dedicated to a specific mobile platform	Using web technologi-es such as HTML, CSS and JavaScript (JS)	Using a native JavaScript bridge to communic-ate between mobile OS and web technology	Using App shell (HTML, CSS, JS), Manifest and Service workers
Installat-ion	App store	Through a URL	App store	Install banner on the webpage
Respons-e Time	High performan-ce	Lesser performan-ce in compariso-n to native apps	Lesser performanc-e in comparison to native apps	Service workers allow smooth performa-nce
Offline support	Partial	None	None	Available
Access to device native features	Yes	No, as the app processes in the server	Yes	Partial
Commo-n languag-es used to develop	iOS: Objective-C or Swift; Android: Java or Kotlin	HTML, CSS, JS	Ionic Framework, Apache Cordova (PhoneGap)	HTML, CSS, JS
Security	Not deployed on a secure layer	Not considered secure as it can also transfer sensitive informatio-n over HTTP	Not deployed on a secure layer	High level of security as it can only be accessed through HTTPS
Cross-platfor-m function-ality	Not available	Available with limited functionali-ty	Available	Available
Backgro-und Synchro-nisation	Available	Available	Available	Available
Push notificat-ions	Can be enabled	Can be enabled	Can be enabled	Can be enabled

Table 1: A comparison of various feature among the Native, Mobile Web application, Hybrid and PWA

1.2 Literature Study

1.2.1 Existing PWA Frameworks

Iadanza et al., (2021) developed a national web platform using PWA to avail the Italian Landslide Inventory (IFFI) data, maps, and reports offline with a geospatial database built on PostgreSQL with PostGIS extension. Some issues faced during development were rethinking the usage in online/offline and the mechanisms to cache resources and map data of the application. Young et al., (2022) developed an application for residents to report the malfunctioning of public facilities to the local government to improve the feedback mechanism in the government. The study concludes by mentioning the need for user testing, app enhancement and scalability in future work. Shahid et al., (2023) present RasterJS, a hybrid client-side web library built using the PWA approach for geospatial data processing. This web library was developed to perform geospatial tasks efficiently using client-side web technologies.

A PWA in healthcare delivery is a prominent use case as it would assist the user with several factors such as accessibility, cross platform application development, offline functionality and responsive design (Iadanza et al., 2021). Rêgo et al., (2019) discussed the need for PWA in the healthcare along with the development of one for evaluation of a patient's diet and highlighted its use as a reliant mobile application that overcomes the challenges of network connectivity. Bin Azhar & Mohan (2022) report the development of a PWA with the advantage of the application being platform-independent and offering access to up-to-date healthcare information, real-time communication between doctors and patients. Delivering of personalized healthcare facility location information can improve and accelerate the decision-making of the user in medical situations and issues (Boulos, 2003).

1.2.2 Location-Based Services in Healthcare

Alexander et al. (2021) presented an Android mobile application to map healthcare facilities in Tanzania and Kenya border catering to the residents' needs in areas irrespective of the network conditions. A limitation of the application was the inability of the users to track the healthcare facility in absence of Wi-Fi. Mushonga et al. (2017) developed a web portal for healthcare facilities and used this information to support the public health administrators in decision-making and public users for spatial information on healthcare facilities.

The usage of PWA in delivery of health care is a novelty and a need with its beneficial capabilities and features (Fernando, 2022; Mole, 2020; Rêgo et al., 2019). From literature, there is a need for a PWA framework with a cross-platform (Mushonga et al., 2017) and offline functionality (Alexander et al., 2021) for

https://doi.org/10.5194/isprs-annals-X-5-W2-2025-521-2025 | © Author(s) 2025. CC BY 4.0 License.

spatial and non-spatial data with data synchronisation in offline / online conditions. Further, there is need for a mobile health application that allows users to select the nearest healthcare facilities from the existing facilities in the database based on their requirements to improve accessibility.

2. Material and Methods

2.1 Study Area

The study area chosen for the POC is Dehradun city in Uttarakhand, India as shown in Figure 2. Dehradun is the capital city of Uttarakhand. It is a major urban area in the state with three of the growing urban centres within 30-50 km of distance, namely Haridwar, Rishikesh and Mussoorie. The city has a population of 569,578 with male and female population, 298,638 and 270,940 respectively (Census, 2011). Due to its small area, a growing population with a mixture of urban and rural settlements, Dehradun makes an ideal study area.

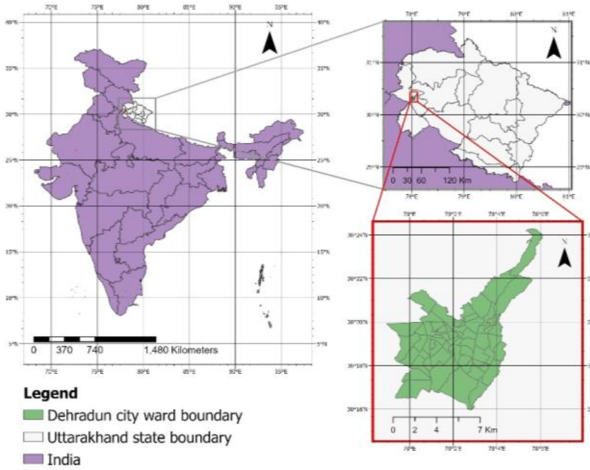

Figure 2 Study area

2.2 Data Sources

For the various healthcare facilities, the coordinates information and healthcare facilities names were obtained from OpenStreetMap using the QuickOSM Plugin (Etienne Trimaille, 2021) in QGIS (OpenStreetMap contributors, 2017). The attribute information such as the address of the healthcare facilities and the services provided is obtained from Government Open Data. Health care facilities were categorised into five categories viz. dental clinic, diagnostic centre, general hospital, multi-speciality hospital and ophthalmology clinic (Figure 3).

2.3 Workflow

The workflow of the study, illustrated in Figure 4, begins with the data preparation and the setup of the PWA. Essential mapping libraries, CSS files and icons are downloaded in the local storage and added to the application's root folder. The literature review aids in formulating the framework layers and their interaction under various network conditions. After the conceptual framework development, the PWA environment is established for practical implementation.

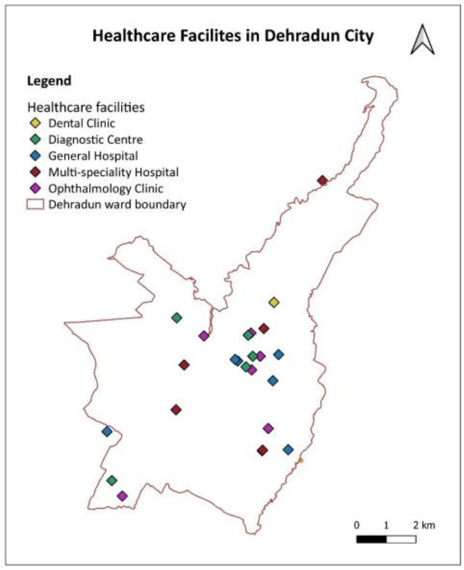

Figure 3 Map representing the health-care facilities based on the services provided

The data from OpenStreetMap and Open Government Data is prepared for data ingestion in QGIS resulting in a GeoPackage including the healthcare facilities POIs and the Dehradun ward boundary layer (Figure 3). GeoPackage serves as a portable database for raster and vector data that operates efficiently in low-connectivity areas (OGC, 2024). For the basemap in the application, map tiles are used. They were first introduced by google with an idea to break the map into smaller parts or "tiles" instead of rendering the entire map each time the user zooms (Forrest, 2023). Map tiles can be classified into vector tiles and raster tiles. Raster tiles are generated using OpenStreetMap XYZ tile and added to GeoPackage, as they pre-load for offline use and require less rendering power (Forrest, 2023; Martin Elias, 2023). JavaScript libraries such as Leaflet (https://leafletjs.com) and Turf.js (https://turfjs.org) facilitate mapping and spatial analysis while the Geopackage library (https://github.com/ngageoint/geopackage-js) allows efficient management of GeoPackage files in web applications (NGA, 2024).

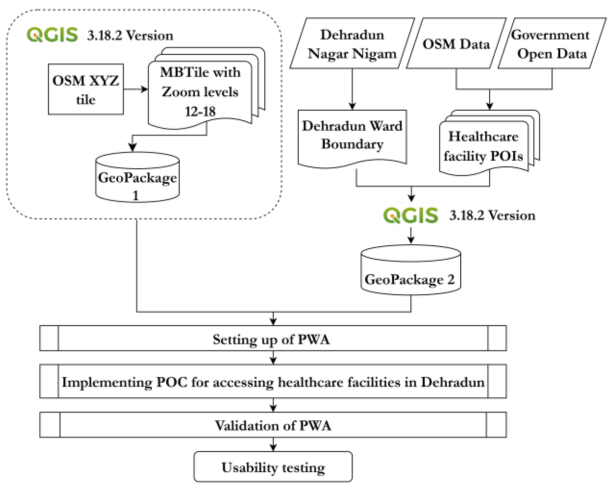

Figure 4 Workflow of the study

The application is hosted on Google Firebase and validated using Google's lighthouse tool. Google's Lighthouse tool was developed to test how well the application is in line with the PWA principles (Correia et al., 2021). Google Chrome no longer supports Lighthouse PWA testing. Hence, webpagetest.org, an open-source tool by catchpoint supporting this tool is utilised as an alternative. Finally, a usability testing is conducted to understand the benefits and shortcomings of the PWA in the real-world. Due to the unavailability of a usability questionnaire specific to a PWA usability, one was devised (Appendix: Usability Test Questionnaire) with the help of Nivala et al., (2007) and Karampanah, (2019) which focus on usability testing of mobile-map based systems.

3. Results

3.1 Conceptual Framework

A conceptual framework paves a path for PWA development from start to finish. It can be adapted and modified based on the requirement (IBM, 2021). The following layers have been identified in the conceptual framework as seen in Figure 5. Throughout the framework, arrows were used to show the PWA's logic flow and the online & offline functionality. The application can communicate requests through the service worker and function offline by retrieving information from the pre-cache.

3.1.1 Data Acquisition, Pre-processing, and Ingestion Layer

The layer constitutes of three sub-layers:
The **data acquisition sub-layer** focuses on gathering the spatial and non-spatial data from various primary or secondary data sources. Once the data is acquired, it is sent to the data pre-processing sub-layer.
In the **data pre-processing sub-layer**, the data is cleaned to avoid duplicate, incomplete, or incorrectly formatted values. After converting into the appropriate data format, it is ingested through the **data ingestion sub-layer** to the database.

3.1.2 Infrastructure Layer

The infrastructure layer supports the underlying technology and resources for PWA operation. It includes layers such as the data layer and application layer functionalities. It also consists of the PWA components necessary for its installation in the user's device: the app shell, the local storage (pre-cache), the service worker and the manifest. It includes client-side components for handling user interactions as well as the data processing and handling the data logic. The data from the data acquisition, pre-processing, and ingestion layer is ingested into the data layer of the infrastructure layer.

3.1.3 Data Layer

The data layer focuses on the data ingested into the application. It deals with the retrieval and managing of the information in the client-data stores from the server (Shahid et al., 2023). The layer communicates with the application layer for the processing of queries and the synchronisation of data in the background. The data synchronisation takes place when the application has an active network.

3.1.4 Application Layer

The application layer consists of the background application logic. This layer includes spatial and non-spatial query processing and the data synchronisation sub-layers. In the query processing sub-layer, the query request from the PWA's client-side is processed and executed while interacting with the data layer. The data synchronisation layer focuses on the data updating in the client-side local database with any changes in the backend database. In offline condition, the PWA ensures that any updating/changes in the data are also reflected in the mobile application's local database. The application layer communicates through a service worker to the presentation layer.

3.1.5 Presentation Layer

The presentation layer consists of information displayed on the user-interface of the mobile application (Richards, 2022). It includes the visualisation of the base map, the geolocation of the user and the POIs, the GIS tools, and the option for users to perform spatial and non-spatial queries including updating the application or the database in case of network availability.

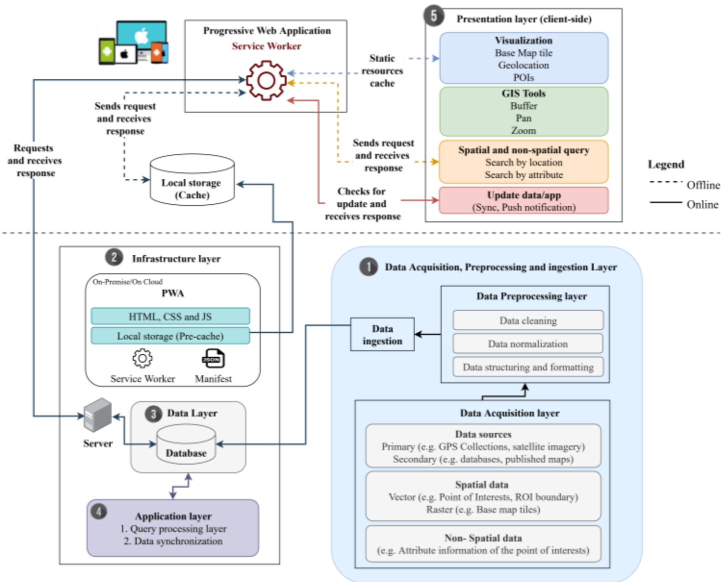

Figure 5 Conceptual Framework

3.2 PWA Proof-of concept - DoonHealth

The PWA developed as POC with the help of the conceptual framework will be referred to as "DoonHealth" henceforth. In this sub-section, the process of implementation of the conceptual framework in each layer in DoonHealth is described.

3.2.1 Framework elements

3.2.1.1 Data Acquisition, Pre-processing, and Ingestion Layer

The first step is the ingestion of data through the data acquisition, pre-processing, and ingestion layer. Two different GeoPackage files are created: one for the Point of Interests (POIs), the healthcare facilities and the boundary layer of the study area and the other for the raster basemap tile to reduce initial loading time and to have the ability to easily update, query and manage the data.

3.2.1.2 Infrastructure Layer

The infrastructure layer of DoonHealth comprises of:

1. **Resources that make up the PWA**: the app shell, service worker and the manifest
2. **Underlying technology**: the JavaScript libraries and CSS files that help the working of the application

3. Data and Application layers

3.2.1.3 Data Layer

In the data layer, the GeoPackage files are ingested in the form of assets into the application through the root folder. When a network request is passed through a service worker, it passes through the application layer and communicates with the data layer to provide a response to the presentation layer. In DoonHealth, Cache API is utilized to load the network resources and the file-based content necessary for the application to provide quick responses regardless of the network condition (Pete LePage, 2020). It follows a cache-first, falling back to network strategy . This strategy aids the application to work in offline conditions as the method precaches the assets and attempts to fetch the requested resource from the cache first, and then checks the server.

3.2.1.4 Application Layer

The spatial query layer logic is supported by turf.js JavaScript library. The non-spatial query layer logic is supported by the jQuery DataTable library (https://github.com/DataTables/DataTables). The data synchronisation sub-layer checks the cached resources and their versions from the data layer and with the help of service worker updates the asset data of the application. The user is notified by displaying a message on the screen in case there is an update as shown in Figure 6.

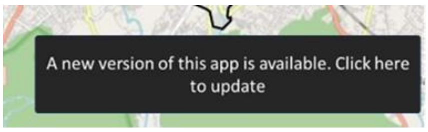

Figure 6 Update message

3.2.1.5 Presentation Layer

The presentation layer focuses on the user-interface of DoonHealth, described in Figure 7. It comprises of the visualisation of the healthcare facilities, the base map, the geolocation, various controls to support application use, spatial and non-spatial data query features.

The user-interface of the DoonHealth:

1. **Zoom controls** –zoom in and out
2. **Zoom Home** – recentre to home location
3. **User location** – view user's current location.
4. **Search by location** - search for the healthcare facilities within a buffer radius specified by user from their current location.
5. **Search by attribute** - search for the healthcare facilities based on their medical requirement.
6. **Legend** – information about the markers in the map.
7. **Layers** – visualization of basemap and healthcare facilities.
8. **Offline/Online** – application offline/online indicator.

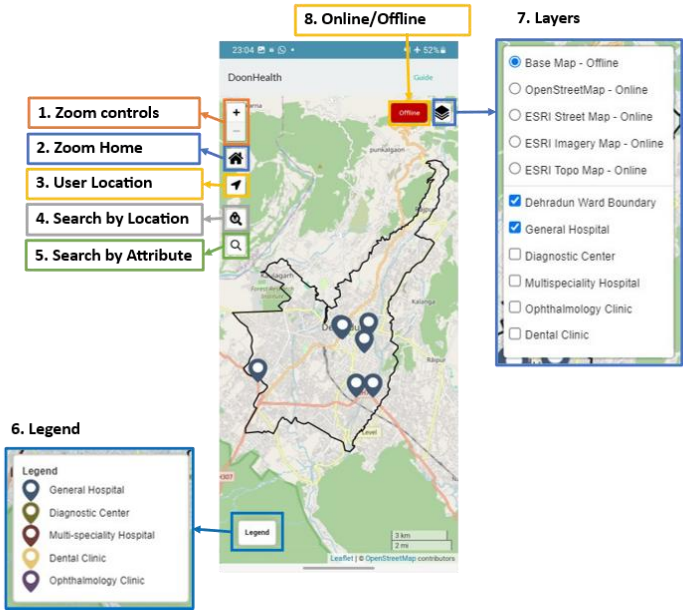

Figure 7 User-interface of DoonHealth

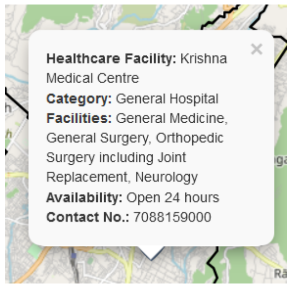

Figure 8 Pop-up information about the facilities.

Each marker provides information such as name of the healthcare facility, its category and the information on availability and contact number as shown in Figure 8.

3.2.2 Tools

The two main tools available in the application for accessing healthcare facilities:

3.2.2.1 Search by location:

This feature allows users to search the facilities within a buffer radius specified from their location. The user can select the category of facility and the type of facility they require. An example for finding the Ophthalmology clinics within a radius of 6 km has been shown in Figure 9.

3.2.2.2 Search by attribute:

This feature allows the user to search from the attribute information available for the data points. The users can use the 'Clear Markers' button to remove the markers added from the search as shown in Figure 10.

3.3 Testing

DoonHealth was tested both using the Google Lighthouse tool to evaluate its compliance as a Progressive Web Application and through an on-ground usability test conducted via a Google form that included a structured questionnaire. The usability test involved 14 participants, with diverse technical backgrounds and device types, primarily Android users aged 25–34.

Participants were asked to perform key offline tasks such as finding nearby healthcare facilities, accessing facility information, and providing feedback.

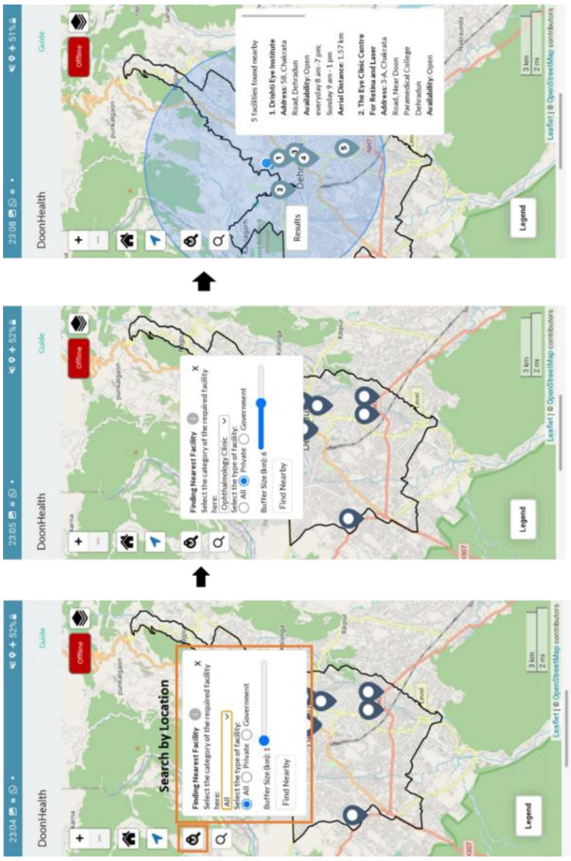

Figure 9 Search by location functionality

The questionnaire covered installation, healthcare-specific features, offline connectivity, adaptability, and overall satisfaction. Participants successfully completed most tasks, though some experienced initial installation challenges due to device restrictions or location permission settings, particularly on iOS. Users provided valuable feedback, suggesting additions like navigation to facilities, ride service integration, and appointment booking features. Overall, the application received positive satisfaction ratings, showcasing its potential as a reliable tool for accessing location-based healthcare services.

4. Discussion

The results of the study highlight the effectiveness of a conceptual framework in guiding the development of a geospatial PWA, DoonHealth in this case, a PWA designed for healthcare delivery. By managing geospatial data through GeoPackage storage and ensuring seamless offline - online transitions, DoonHealth meets critical user needs in remote or unfamiliar neighbourhoods. Healthcare seekers, particularly those requiring admission or seeking specific levels of care, benefit from the app's ability to locate facilities, view attributes (such as specialty services), and access basemap data without network connectivity. The conceptual framework's strength lies in its cross-platform compatibility and modular design, which guarantee accessibility, affordability (install via URL), and acceptability across Android and iOS devices. Usability testing revealed high user satisfaction with the app's performance and offline functionality, though initial installation restrictions and occasional caching delays were noted.

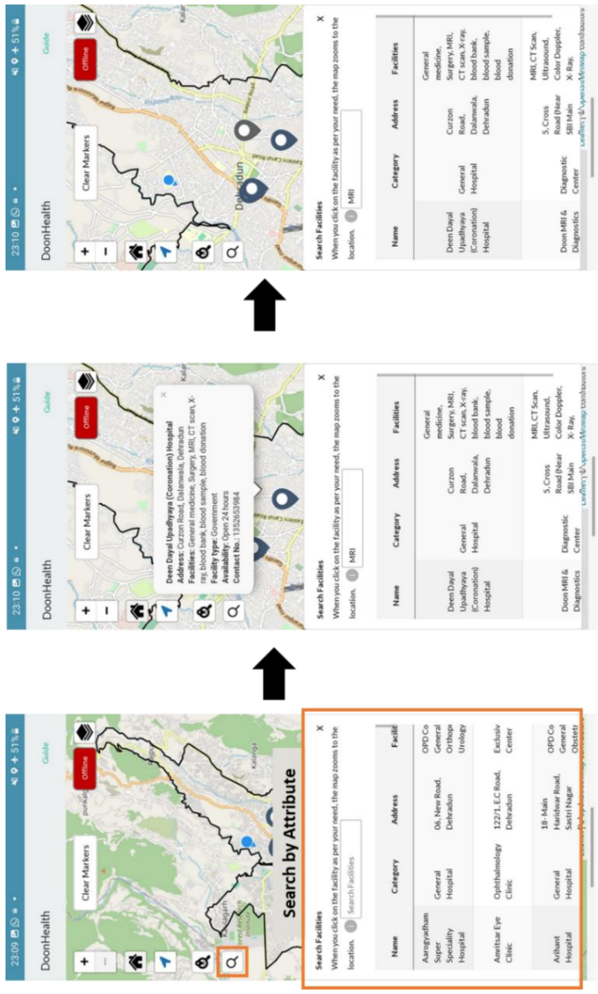

Figure 10 Search by attribute feature

Some limitation of DoonHealth PWA include:

- Device restrictions on home-screen installations reduced ease of use for some participants.
- Incomplete asset caching in low-bandwidth scenarios led to temporary sluggishness.
- The current absence of routing, real-time facility status, and integrated appointment or transport booking limits end-to-end patient journey support.

Including input from healthcare providers can also help tailor facility metadata (e.g., specialities offered, hours of operation) to improve decision making at the point of need. Beyond healthcare, this conceptual framework readily adapts to other domains: tourism (offline guides to attractions), public transit (route schedules and stops), and emergency response (incident mapping and resource allocation). Each of these use cases illustrates how the conceptual framework can be adapted to meet diverse user needs across different sectors, thereby promoting a broader adoption of PWAs in various domains. The novelty of this study lies in developing and applying a unified conceptual framework for geospatial PWA, taking a healthcare context as a proof-of concept to demonstrate GeoPackage-backed offline storage and progressive enhancement can support spatial and non-spatial data handling.

5. Conclusion and Future Directions

The conceptual framework designed in this study serves as a model for future cross-platform geospatial PWAs. By combining spatial and attribute queries with offline basemap and facility data, the application ensures uninterrupted access to essential services. To realize the next phase, development efforts should focus on Implementing client-side routing algorithms and Integrating live data feeds for real-time updates. These steps will drive geospatial innovation in PWA development and pave the way for user-centric geospatial applications in the future.

Acknowledgements

I would like to thank Mr. Kapil Oberai, Dr. Carmen Anthonj and Dr. Rob Lemmens for their immense patience and support through this journey. I would also like to thank the scientists and faculty members from the Indian Institute of Remote Sensing (IIRS) and the Faculty of Geo-Information Science and Earth Observation (ITC) who have constantly inspired me and have significantly contributed to my academic growth over the past two years.

References

Alexander, H., Leo, J., & Kaijage, S. (2021). Online and Offline Android Based Mobile Application for Mapping Health Facilities Using Google Map API. Case Study: Tanzania and Kenya Borders. *Journal of Software Engineering and Applications*, *14*(08), 344–362. https://doi.org/10.4236/jsea.2021.148021

Bin Azhar, M. A. H., & Mohan, J. T. (2022). Progressive Web App for Real-time Doctor-Patient Communication and Searchable Health Conditions. *10th E-Health and Bioengineering Conference (EHB)*, 1–5. https://doi.org/10.1109/EHB55594.2022.9991288

Boulos, M. N. K. (2003). Location-based health information services: A new paradigm in personalised information delivery. *International Journal of Health Geographics*, *2*(1), 2. https://doi.org/10.1186/1476-072X-2-2

Charkaoui, S., Adraoui, Z., & Benlahmar, E. H. (2015). Cross-platform mobile development approaches. *Colloquium in Information Science and Technology (CIST)*, 188–191. https://doi.org/10.1109/CIST.2014.7016616

Correia, F., Ribeiro, Ó., & Silva, J. C. (2021). Progressive Web Apps Development: Study of Caching Mechanisms. *2021 21st International Conference on Computational Science and Its Applications (ICCSA)*, 181–187. https://doi.org/10.1109/ICCSA54496.2021.00033

Das, R. C., & Alam, T. (2014). Location based emergency medical assistance system using OpenstreetMap. *2014 International Conference on Informatics, Electronics & Vision (ICIEV)*, 1–5. https://doi.org/10.1109/ICIEV.2014.6850695

Dolan, B. (2012, February 14). *12 trends for mobile health in 2012*. Mobi Health News. https://www.mobihealthnews.com/16310/12-trends-for-mobile-health-in-2012

Etienne Trimaille. (2021, September). *QuickQSM Plugin*. https://plugins.qgis.org/plugins/QuickOSM/version/2.0.0/

Fernando, K. (2022). *Development of Smart Healthcare Web System using PWA with AI: A Review of Related Literature*.

Forrest, M. (2023, January 23). *Map Tiles: Everything You Need To Know* [Blog]. Map Tiles: Everything You Need To Know. https://carto.com/blog/map-tiles-guide

Frankston, B. (2018). Progressive Web Apps [Bits Versus Electrons]. *IEEE Consumer Electronics Magazine*, 106–117. https://doi.org/10.1109/MCE.2017.2776463

Iadanza, C., Trigila, A., Starace, P., Dragoni, A., Biondo, T., & Roccisano, M. (2021). IdroGEO: A Collaborative Web Mapping Application Based on REST API Services and Open Data on Landslides and Floods in Italy. *ISPRS International Journal of Geo-Information 2021, Vol. 10, Page 89*, *10*(2), 89. https://doi.org/10.3390/IJGI10020089

IBM. (2021, March 5). *Relational database structure—IBM Documentation*. https://www.ibm.com/docs/en/mam/7.6.0?topic=design-relational-database-structure

ITU. (2023). *Measuring digital development* (Facts and Figures). International Telecommunication Union. https://www.itu.int/hub/publication/d-ind-ict_mdd-2023-1/

Jabareen, Y. (2009). Building a Conceptual Framework: Philosophy, Definitions, and Procedure. *Int. J. Qual. Methods*, *8*. https://doi.org/10.1177/160940690900800406

Karampanah, S. (2019). *A Survey of Usability Issues in Mobile Map-based Systems* [University of Münster]. https://run.unl.pt/bitstream/10362/67711/1/TGEO0205.pdf

Kratzke, C., & Cox, C. (2012). Smartphone Technology and Apps: Rapidly Changing Health Promotion. *International Electronic Journal of Health Education*, *15*, 72–82.

Martin Elias. (2023, September). *Raster vs Vector Map Tiles: What Is the Difference Between the Two Data Types?* Maptiler Documentation Portal.

MDN Contributors. (2023). *Making PWAs work offline with Service workers—Progressive web apps*. Mozilla Developer. https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps/Tutorials/js13kGames/Offline_Service_workers

Mole, P. (2020). *Progressive Web Apps: A Novel Way for Cross-Platform Development*.

Mushonga, H. T., Banda, F., & Mulolwa, A. (2017). Development of a web based GIS for health facilities mapping, monitoring and reporting: A case study of the Zambian Ministry of health. *South African Journal of Geomatics*, *6*(3), Article 3. https://doi.org/10.4314/sajg.v6i3.4

NGA. (2024, May 17). *Ngageoint/geopackage*. Npm. https://www.npmjs.com/package/@ngageoint/geopackage

Nivala, A.-M., Tiina Sarjakoski, L., & Sarjakoski, T. (2007). Usability methods' familiarity among map application developers. *International Journal of Human-Computer Studies*, *65*(9), 784–795. https://doi.org/10.1016/j.ijhcs.2007.04.002

OGC. (2024, June 2). *GeoPackage*. https://www.geopackage.org/spec140/

Oluwatofunmi Adetunji, Chigozirim Ajaegbu, Nzechukwu Otuneme, & Olawale J. Omotosho. (2020). Dawning of Progressive Web Applications (PWA): Edging Out the Pitfalls of Traditional Mobile Development. *American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)*, *68*(1), 85–99.

OpenStreetMap contributors. (2017). *OpenStreetMap*. Https://Www.Openstreetmap.Org/. https://www.openstreetmap.org/#map=11/30.3628/77.9453

Patrick, K., Griswold, W. G., Raab, F., & Intille, S. S. (2008). Health and the Mobile Phone. *American Journal of Preventive Medicine*, *35*(2), 177–181. https://doi.org/10.1016/J.AMEPRE.2008.05.001

Pete LePage. (2020, April 27). *Storage for the web*. Web.Dev. https://web.dev/articles/storage-for-the-web

Rêgo, F., Portela, F., & Santos, M. F. (2019). Towards PWA in Healthcare. *Procedia Computer Science*, *160*, 678–683. https://doi.org/10.1016/j.procs.2019.11.028

Richards, M. (2022). *Software Architecture Patterns*. O'Reilly Media, Inc. https://www.oreilly.com/library/view/software-architecture-patterns/9781491971437/ch01.html

Shahid, M., Sermet, Y., Mount, J., & Demir, I. (2023). Towards progressive geospatial information processing on web systems: A case study for watershed analysis in Iowa. *Earth Science Informatics*, *16*(2), 1597–1610. https://doi.org/10.1007/S12145-023-00993-X/TABLES/2

Steiniger, S., Neun, M., & Edwardes, A. (2006). Foundations of Location Based Services. In *CartouCHe—Lecture Notes on LBS*. https://www.researchgate.net/publication/230777155_Foundations_of_Location_Based_Services

Young, R. L. C., Meskaran, F., & Eaganathan, U. (2022). *Public Facilities Feedback Application: A Location-based Solution of Malaysian public Facilities Issues*. 1–5. https://doi.org/10.1109/MysuruCon55714.2022.9972588

Appendix: Usability Test Questionnaire

Introduction

1. Name:
2. Gender:
3. Age:
4. Device type: Mobile / Tablet / Desktop
5. Do you have a background in Geo-information Sciences? Yes / No
6. Device OS: Android / iOS
7. In case of iOS, please mention the model of your iPhone and iOS version:

Section 1: Tasks

After installing the application, it is advisable to perform the following tasks after switching your device to the airplane mode to explore the offline features and then answer the questions in the second section.

Task 1: Finding Nearby Healthcare Facilities

Imagine you need to find the nearest hospital or healthcare clinic to your current location using DoonHealth. Please locate and identify the closest healthcare facility.

In case of completion with some observations, please use the "Other" section to answer.

Options: Completed / Unable to complete / Other:

Task 2: Using Offline Features in Low Connectivity Areas

Imagine you are in an area with limited internet connectivity. While offline, try to access DoonHealth's features, such as locating the facility and viewing the necessary additional information about the facility.

Options: Completed / Unable to complete / Other:

Task 3: Providing Feedback or Reporting Issues

As a user of the application, you might encounter a problem or have feedback to share. Submit your feedback or report the issue here.

Section 2: Questionnaire

After performing the tasks, answer the following questions based on your understanding of the application.

2.1 Installation

2.1.1. Did DoonHealth provide clear instructions on how to install and add it to the home screen?

Options: Yes / No / Other:

2.1.2. Was the application's icon visible and easily accessible on the home screen after installation?

Options: Yes / No / Other:

2.2. Healthcare-Specific Features

2.2.1. Were you able to search for healthcare facilities (e.g., hospitals, clinics) based on your location or specific criteria within the application?

Options: Yes / No / Other:

2.2.2. Did DoonHealth provide necessary and accurate information for accessibility of the healthcare services?

Options: Yes / No / Other:

2.3. Connectivity

2.3.1. Did the application provide a seamless experience when transitioning between online and offline modes, ensuring uninterrupted access to essential healthcare information?

Options: Yes / No / Other:

2.3.2. Were you able to access location-based healthcare services and resources even when offline?

Options: Yes / No / Other:

2.3.3. After installing and using the app, have you noticed any significant changes in your device's battery life?

Options: Yes / No / Not sure or Have not noticed / Other:

2.4. Adaptability and Simplicity

2.4.1. Did DoonHealth adapt well to the screen size and orientation of your device, ensuring readability and usability?

Options: Yes / No / Other:

2.4.2. Were the visual design elements (e.g., color, fonts) consistent throughout, providing a cohesive user experience?

Options: Yes / No / Other:

2.4.3. Did the navigation structure remain consistent across different sections of DoonHealth, facilitating seamless access to healthcare-related features?

Options: Yes / No / Other:

2.5. Overall Satisfaction

2.5.1. On a scale of 1 to 5, how satisfied are you with your experience using DoonHealth for accessing location-based healthcare services?

1 - Very Dissatisfied; 5 - Very Satisfied

2.5.2. Would you recommend DoonHealth for accessing healthcare services to others? Why or why not?

2.6. Additional Feedback

2.6.1 Please provide any additional comments, suggestions, or concerns you have regarding the usability of the application, particularly in relation to location-based services for healthcare.

2.6.2. In case you work as a healthcare professional, could you please provide any additional comments, suggestions, or concerns that can make this application better?