Nitte Meenakshi Institute of Technology 

(AN AUTONOMOUS INSTITUTION AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY BELGAUM)

**(A Unit of Nitte Education Trust, Mangalore)**

**PB No. 6429, Yelahanka, Bangalore 560-064, Karnataka**

# “A Case Study of Application Development for Mobile and Location-Based Services”

*Submitted in partial fulfilment of the requirements for the award of the degree of*

MASTER OF COMPUTER APPLICATIONS

of

Visvesvaraya Technological University (VTU)



By

# ROHIT KUMAR RAY – 1NT22MC084

Under the guidance of

MR. NAVNEETH A.V

Asst. Professor

Academic Year: 2023 – 24

 **Nitte Meenakshi Institute of Technology** 

**Yelahanka, Bengaluru- 560064**

**Department of Master of Computer Applications**

# CERTIFICATE

*This is to certify that* ***ROHIT KUMAR RAY*** *bearing USN* ***1NT22MC084*** *has completed the Learning Activity as a part of II Semester Curriculum,* ***Mobile Application LA-2*** *during the academic year 2022-2023.*

## Signature of faculty In-charge Signature of HOD

MR. NAVNEETH A.V Dr. Sreekanth R.

Professor Prof & Head, Dept. of MCA

NMIT, Bengaluru NMIT, Bengaluru

**A Case Study of Application Development for**

**Mobile and Location-Based Services**

**ROHIT KUMAR RAY**

Department Of Master of Computer Application,

Nitte Meenakshi Institute of Technology (NMIT) rohitbhaicpr8485@gmail.com

## Abstract

In this essay, we give a case study of a location-dependent service that caters to visitors. The development of two applications, one based on native technology, the other on HTML5, and frames that are linked. We give implementation information and compare the various options for location support, compatibility across platforms, and compliance with essential features. Our research demonstrates that web-based strategies may have substantial advantages over using native technology; both applications showed a similar level of location support and feature compliance. even if the web version replaced the native version in a cross-platform capability.

This case study details the process of creating "Enhance and Explore," a smartphone app that uses location-based services to provide travelers with tailored recommendations and easy navigation. Insights into the ideation, planning, design, development, testing, deployment, and post-launch tactics used to produce an engaging mobile experience are provided by the study. This case study highlights the potential for location-based services to improve travel experiences and depicts the road to prosperous mobile application development in this field by putting an emphasis on user-centric design, strong technological implementation, and efficient marketing.

### KEYWORDS

Network services, Location-based services, App Store Submission, Server

Deployment, Monitoring and Maintenance, User Acquisition **INTRODUCTION**

Access to data is made possible everywhere, at any time, by mobile computing. The nature of the accessed data might be extremely diverse, including static textual material, connections to audio or video (multimedia sessions), mobile eservices, or background data obtained by sensors or GPS. One of the most crucial context elements in mobile computing is location. The location of the user is especially important. The system's ability to give personalized answers to each user can be improved by informing it of the user's current location. [11] Consider Applications that provide discounts to users based on the stores they are close to, emergency services, or navigation services that come from the user, as opposed to similar services that don't offer these features Without charge, you are permitted to print or copy all or a portion of this work for your own personal or educational use, provided that copies are made with this notice and the whole citation on the first page and are not made or disseminated for profit or commercial gain. Any additional copies, reprints, server posts, or list distributions require prior authorization and/or payment. Examples are programs that provide customers with discounts based on the stores they are close to or emergency location support. The former is obviously superior to the latter since it has the ability to provide a fuller service with greater accuracy and reality. Thus, locationbased services (LBS) and location-dependent applications (LDA) are crucial components of our current computing environment. There is much to consider when developing LBS for mobile devices. Mobile devices in general have limitations in terms of e.g. screen size, battery, and network interfaces. Also, mobile devices come in many shapes, manufactured by different vendors, equipped with different hardware, and running different operating systems. There are also often big dissimilarities between older and more recent versions of the same operating system, both in terms of functionality and which applications they support. When developing mobile applications, two main alternatives exist regarding implementation technology: either you go for native applications, or you deliver mobile web in the shape of web apps.

### RELATED WORK

Examples of LBS implementations include GeoHealth [4], an interactive service for supporting distributed mobile collaboration in the healthcare industry. relying on web technology Using technology, GPS positioning, and Google Maps, GeoHealth provides real-time task interchange, fast alarm capabilities, and location- and user-specific information. Using technology, GPS positioning, and Google Maps, GeoHealth provides real-time task interchange, fast alarm capabilities, and location- and user-specific information. A web service architecture known as Person Wide Web [12] recognizes web resources based on the user's location. The performance of the architecture is assessed using an Android application that shows service setup and message delivery times. A location-based game called Tidy City is presented by Wetzel, Blum, and Oppermann [16] and enables locals and visitors to explore a city using a locationsensitive application. In order to provide services, a hybrid method combining both web-based interfaces and mobile technologies is used. Personalized LBS delivery and online service utilization are made possible by the Lewis system [9]. Lewis can provide recommendations for local locations, including those based on social networks, when the user is on the go. Mobile tourism—the use of mobile devices as tour guides—is a topic covered by Kenteris, Gavalas, and Economou [8]. Their prototype complies with user preferences, is portable, and provides extensive information. It doesn't, however, have complete cross-platform support. The value of mobile applications for LBS has been demonstrated, and it cannot be overstated. Mobile application development can be divided into two primary categories: web-based applications or applications produced using native programming languages [13] [2]. Commonly emphasized are a few general characteristics of native vs. online. For a certain set of devices, native apps are made. Such apps follow the design rules relevant to the target device and can utilize its hardware characteristics, such as the operating system and camera. The native app is often obtained from a digital storefront run by the company that developed the operating system, such as Apple's AppStore or Google's Google Play. The web app, on the other hand, is just a website that has been modified for mobile web browser access, including changes to screen size, cellular network speed, and mobile navigation controls.

METHOD

When it comes to the development of location-dependent applications, we conducted a case study to examine how effectively cutting-edge web technologies can match native approaches. Giving a comprehensive account of how an application is used in a real-world situation is the primary objective of a case study [10]. The explanation can be used to derive inferences from the particular situation and evaluate the viability of applying them to a given area, in this case, application development for mobile and LBS. For example, Walsham [15] discusses this type of interpretive research approach. The notion to establish a mobile app at a tourist attraction in Gammelstad, Lule, a northern region, came about as part of the ongoing research project TG4NP2. A concept to build a smartphone application that would enable visitors to a tourist center in Gammelstad, Lule in northern Sweden to engage in outdoor tourist attractions even when there is no personnel present arose as part of an ongoing research project named TG4NP2. The Gammelstad location is a historic church town from the fourteenth century that has a lot of attractions nearby. The tourist information centre is normally staffed during the day, and guided tours are periodically offered in a variety of languages. According to the tourist center's owners, many visitors may arrive in the evening when services are unavailable. Using only the technology included in a mobile device like a smartphone, one of the objectives of the program is to offer an experience that comes as near to a guided tour as feasible. The app should be able to give instructions to different public amenities including restrooms, parking lots, and restaurants, which is another feature that the project owner wants it to have. One of the needed features was a thorough map with properly highlighted points of interest or POIs. Textual information about a particular POI will be presented, maybe combined with an audio description, as Auser gets closer to it. pictures or video recording. It will be simple for any visitor with a smartphone to access by posting a banner outside the visitor center with a QR code that can be scanned in order to load the program and brief instructions on how to use it.

### RESULTS

The software architecture's main goal was to make a significant separation between content and functionality. The app's actual informational contents could be expressed in a format that is simple to change and tailored to fit a wide range of uses. We went with the XML-based Keyhole Markup Language (KML) for its file format. It offers easily understood and editable text files that are humanreadable. Additionally, the format enables a content provider to build several informative item layers, making it simple to abstract away unwanted information. With the KML file format, HTML-coded items like text, graphics, and embedded media can be added to geographical information objects like placemarks or arbitrary polygons expressed with geographic coordinates. This might be demonstrated by the fact that general-purpose items like restaurants and be kept off the map until the user decides to show them. As a result, they won't obstruct the primary interest. KML files can either be created using a specialized editor where objects are drawn on a map and extra contents are added in an integrated HTML editor, or they can be edited directly in a text editor. The KML file can be updated quickly and its contents can be handled dynamically by being stored on a central server. The elements Folder, Placemark, Name, Point, and (optional)

Description are included in the KML file. On the map, layers of placemarks are created using folder elements. There is a name element and a Point that indicates the position in each Place-mark element. A description element can also be inserted to offer material for a popup window that shows the user rich HTMLbased content. Sub-groups of information items can be created using the themes element and displayed when clicked in a popup window. A collection of fundamental features is offered when it comes to the app's fundamental functionality. A main page containing a map view component that shows data from a map provider, the information from the selected KML file, as well as the user's position, is one example of this. The map view screen is based on the Leaflet API3, which uses JavaScript to display maps and different overlays. Google Maps4, Bing Maps5, Open Street Maps6, and many other map providers are among the many that Leaflet supports.

**Native App:** The native app is implemented using the Android platform and centered around the Google Maps API, which is distributed as part of the Google Play services SDK. The app is thereby limited to using only one map provider. The map component is configured in full-screen mode on the device, and all additional content is shown on top of the map. Content, in the form of a KMLfile, images, and sound clips, are stored on a web server. When the app is loaded, the KML-file is downloaded from the server and parsed using a customized parser. Placemarks are drawn as icons on an overlay on the map. By using the menu button, a dialog can be shown that allows a user to choose which layers of POIs to show on the map and each layer of icons is color-coded. If an internal GPS is present in the device, an icon is shown in the center of the image to indicate the user’s own location. If a POI is within 50 m, or if a POI placemark is clicked by the user, a popup window is shown that displays the HTML-coded information from the KML-file, including images, etc. that corresponds to the chosen placemark.

**HTML5 Web App:** The HTML5 version of the app has many similarities with the native one. The first ma jor difference is that it is based on code that is HTML5-compliant, enabling it to execute on a large variety of devices. Since the application is intended to run in a web browser with JavaScript support, a large number of map API solutions are available. The open-source Leaflet API is chosen since it is specifically designed to support mobile devices. It allows for a variety of map providers, and comprehensive support for zooming, panning, and dragging, and it is well-documented for developers. Also, Leaflet API provides a built-in set of controls, such as markers and popups. However, it was determined during development that the built-in controls did not offer the desired level of flexibility so custom versions were implemented to suit the specific application. Since the popups are likely to contain more than just a few lines of text, there were problems with scrolling and layout for some devices. It turned out to be quite challenging to achieve the same layout and basic functionality on many different platforms and custom handling was required in some cases, especially on older devices. A KML file of the same format as the one used in the native app is loaded from a web server and displayed using color-coded markers. By clicking on an icon in the top right corner, a list is shown that allows the user to choose between layers of interest. The Geolocation API is used to retrieve the user's own location and functionality similar to that of the native app is implemented that allows for automatic popups when the user approaches a POI.

**Comparative Evaluation:** The two prototypes were designed to support the same basic functionality. As the prototypes were developed on different occasions by different programmers, properties such as graphical design and additional features did differ. In our comparative evaluation, we looked solely at the requirements set up for this particular case study, i.e. location support, compliance with required features (see detailed description under section III), and crossplatform functionality. The differences in e.g. GUIs and extra features between the two versions were not studied, and thus such differences have not affected our evaluation in any significant way. When it comes to location support, both versions of the app have comparable functionality. The POI map is implemented in both apps, allowing geographical location representation of important objects. The information is location-aware, e.g. locations already visited by the user are marked with a certain color. Layers are used to e.g. filter out restaurants and restrooms from the POIs. Differences between the app versions are negligible as our evaluation shows that differences in location support and GPS precision come from hardware. Compliance with other required features is also equal between the two different versions of the app. Both support dynamically loaded content, that is easy to maintain for non-technical staff. There is also equal support for multimodal media (e.g. text, links, audio, and video in the POI descriptions).

### DISCUSSION

Application developers now have highly effective tools for offering mobile and LBS to end users thanks to the enormous expansion in terms of both device and network deployments internationally. The mobile app ecosystems, on the other hand, favor software creation and deployment are now incredibly simple to handle thanks to technologies provided by Apple, Google, and Microsoft. On the other hand, advanced web technologies targeted towards mobile and LBS are quickly receiving interest from sizable development communities. There is minimal need for a marketplace when it comes to free, location-based applications that are meant to be used on-site. Users (in our case, visitors) can easily access and begin using the application on-site as long as they are given a URL to it. The use of web-based methodologies may have considerable advantages over employing native technology, as this article has demonstrated. The two most significant benefits of that strategy are being able to reach all users from a single implementation and avoiding deployment challenges connected to certain relevant ecosystems. The disadvantages include dealing with a technology that isn't fully developed and doesn't yet have complete support from all available browsers.

One intriguing aspect of our web-based strategy was that JavaScript technology was used on the client side to parse the KML file. As a result, the web-based implementation became more comprehensive and less reliant on continuous network connectivity.

### CONCLUSION

We have explored the development of a mobile application focused on location-based services. Throughout the project, we have highlighted the key steps and considerations involved in creating such an application, from conceptualization to deployment. The development of mobile and location-based service applications has become increasingly relevant in our modern, hyperconnected world, and this case study has shed light on the challenges and opportunities this domain presents. the development of mobile and location-based service applications is a complex but rewarding endeavour. Success in this domain requires a holistic approach that encompasses user-centric design, robust technology infrastructure, adherence to privacy and security standards, and ongoing efforts to meet user expectations. As technology continues to advance and user demands evolve, staying adaptable and innovative is key to maintaining the relevance and competitiveness of such applications in the ever-changing mobile landscape.

### REFERENCES

1. M. Anttonen, A. Salminen, T. Mikkonen, and A. Taivalsaari. Transforming the web into a real application platform: new technologies, emerging trends, and missing pieces. In Proceedings of the 2011 ACM Symposium on Applied Computing, SAC ’11, pages 800–807, New York, NY, USA, 2011. ACM.
2. K. Buettner and A. Simmons. Mobile web and native apps: How one team found a happy medium. In A. Marcus, editor, Design, User Experience, and Usability. Theory, Methods, Tools and Practice, volume 6769 of Lecture Notes in Computer Science, pages 549–554. Springer Berlin / Heidelberg, 2011.
3. A. Charland and B. Leroux. Mobile application development: web vs. native. Commun. ACM, 54(5):49–53, May 2011.
4. C. M. Christensen, J. Kjeldskov, and K. K. Rasmussen. Geohealth: a location-based service for nomadic home healthcare workers. In

Proceedings of the 19th Australasian Conference on Computer-Human Interaction: Entertaining

User Interfaces, OZCHI ’07, pages 273–281, New York, NY, USA, 2007. ACM.

1. S. Fahlesson and D. Johansson. A case study of cross-platform web application capability. In Future Network and MobileSummit 2013 Conference Proceedings, FNaMS, 2013.
2. N. P. Huy and D. van Thanh. Evaluation of mobile app paradigms. In

Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia, MoMM ’12, pages 25–30, New York, NY, USA, 2012. ACM.

1. S. Ilarri, E. Mena, and A. Illarramendi. Location-dependent query processing: Where we are and where we are heading. ACM Comput. Surv., 42(3):12:1–12:73, Mar. 2010.
2. M. Kenteris, D. Gavalas, and D. Economou. An innovative mobile electronic tourist guide application. Personal Ubiquitous Comput., 13(2):103– 118, Feb. 2009.
3. S.-P. Ma, W.-T. Lee, and C.-H. Kuo. Location Explorer with Information

Services: A mobile application to deliver location-based web services. In NextGeneration Electronics (ISNE), 2013 IEEE International Symposium on, pages 283–286, 2013.

1. M. Q. Patton. Qualitative research & evaluation methods. SAGE, London,

4. ed. edition, 2004.

[11] J. Schiller and A. Voisard. Location Based Services. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.