To:	Gail Hohner, MDP Director Brian Pasmanter, John Policicchio, Sponsors Jeff Ringenberg, Faculty Mentor
From:	MDP 2022 Cohort Team (Bozhou Chen, Andy Ho, Ember Shan, Jae Min Shin, Patrick Su, Donglin Yu, Yuran Zhang)
Sponsor:	MotorCity Casino Hotel
Project Title:	Beacon Driven Casino Floor Navigation in a Mobile App
Date:	February 6, 2022
Subject:	Executive Summary 1

Project Briefing

Expected Deliverable, Goals and Project Scope

By December 2022, the MotorCity Casino Hotel can expect the MDP 2022 Cohort to provide an Android mobile application that can be used by customers and staff to navigate the Casino floor with turn-by-turn directions. With 100,000 square feet spread over two floors, navigation can be challenging for both patrons seeking specific games or services, and staff attempting to deliver food/drinks to a customer quickly. While robust services, such as Google Maps, exist for navigation in outdoor environments, these services lack accuracy and specificity in indoor environments. The expectation is that the app, using existing Bluetooth beacons embedded in slot machines and additional beacons installed by the MDP 2022 Cohort, will allow customers to navigate the Casino floor independently - thereby providing a highly individualized experience for each customer.

Success will be measured based on efficacy of the app, namely: ease of use, accuracy of current location and directions provided. The team will test the app by attempting to navigate the Casino floor in all reasonable use cases – game to game, service to game and navigation across the two different floors. In addition, the team hopes to be able to trial the app with Casino staff and customers to ensure its usability at all levels of technical ability.

Project Context

<u>Frontend / UI / UX</u>

The frontend sub team will create an Android application using Webview that displays the current location of the user’s device on the Casino floor. The interface allows the user to enter a desired destination by selecting a point on the map or choosing a type of game/service from a dropdown menu or using a search bar. Once a destination has been selected, the app will highlight the shortest path from the user’s current location to their destination and continuously update their current location as they proceed to their destination. This sub team will learn how to produce an easy to use interactive mobile app for public use.

<u>Backend / Hardware</u>

The backend and hardware sub teams will implement a SQL database running on the MotorCity Casino’s servers that contains the unique id of each Bluetooth beacon and its location on the Casino floor, an API for the Android app and a pathfinding algorithm. Using the signal strengths and unique IDs of the Bluetooth beacons detected by a user’s device, the database will be used to determine the user’s location. Once a destination has been selected, the pathfinding algorithm will send turn-by-turn directions to the user’s device. The project will require configuring the Bluetooth beacons in the slot machines to provide a signal intensity high enough to allow reliable detection but low enough to prevent erroneous signals from being detected across different floors.

Project Organization

The team is made up of seven members, a faculty mentor and two sponsors. Three sub teams have been formed to focus on different aspects of the development process. The sub teams with their respective members are: Frontend /UI/UX (Ember Shan, Yuran Zhang), Backend (Andy Ho, Patrick Su) and Hardware (Bozhou Chen, Jae Min Shin and Donglin Yu). These sub teams are flexible, as the project requires significant coordination between all aspects of its development. Meetings of sub teams are conducted on an ad-hoc basis and thus do not have specific meeting times. Our student team meetings are held with all members as follows:

Meeting Type	Meeting Time	Attendees
Sponsor and Faculty Meeting	Wednesday @ 5-7 PM EST (Sponsors join from 5-6pm EST)	Sponsor, Mentor, MDP 2022 Cohort
Full Student Team Meeting	Sunday @ 2-4 PM EST	MDP 2022 Cohort

Table 1: Weekly Team Meeting Schedule. This table details the time/attendance of the weekly meetings

The team rotates the responsibility of organizing and leading our weekly sponsor and faculty meeting and the scribe for each meeting using a roster system. The weekly sponsor and faculty meeting is held on Zoom while the full student team meeting and the various sub team meetings are held either on Zoom or in-person on an as-needed basis. While a scribe is designated for each meeting, all members contribute to the meeting minutes pertaining to their sub team for each meeting. Files, such as weekly meeting agendas and follow-ups are shared on a shared Google Drive while code is stored on an Azure DevOps server created and maintained by our sponsor. Our primary form of communication between group members, sponsor and faculty mentor is a Discord server.

The team discusses our short-term tasks and long-term goals midway during our weekly sponsor and faculty meeting to keep our sponsors and faculty mentors up-to-date and accept their input. The exact details and allocations of these tasks and goals are discussed by the student team following the sponsors' and faculty mentor's departure. These tasks and goals are recorded and tracked using the Boards feature of the MotorCity Casino Azure DevOps server at the end of each weekly faculty and sponsor, and full student team meetings.

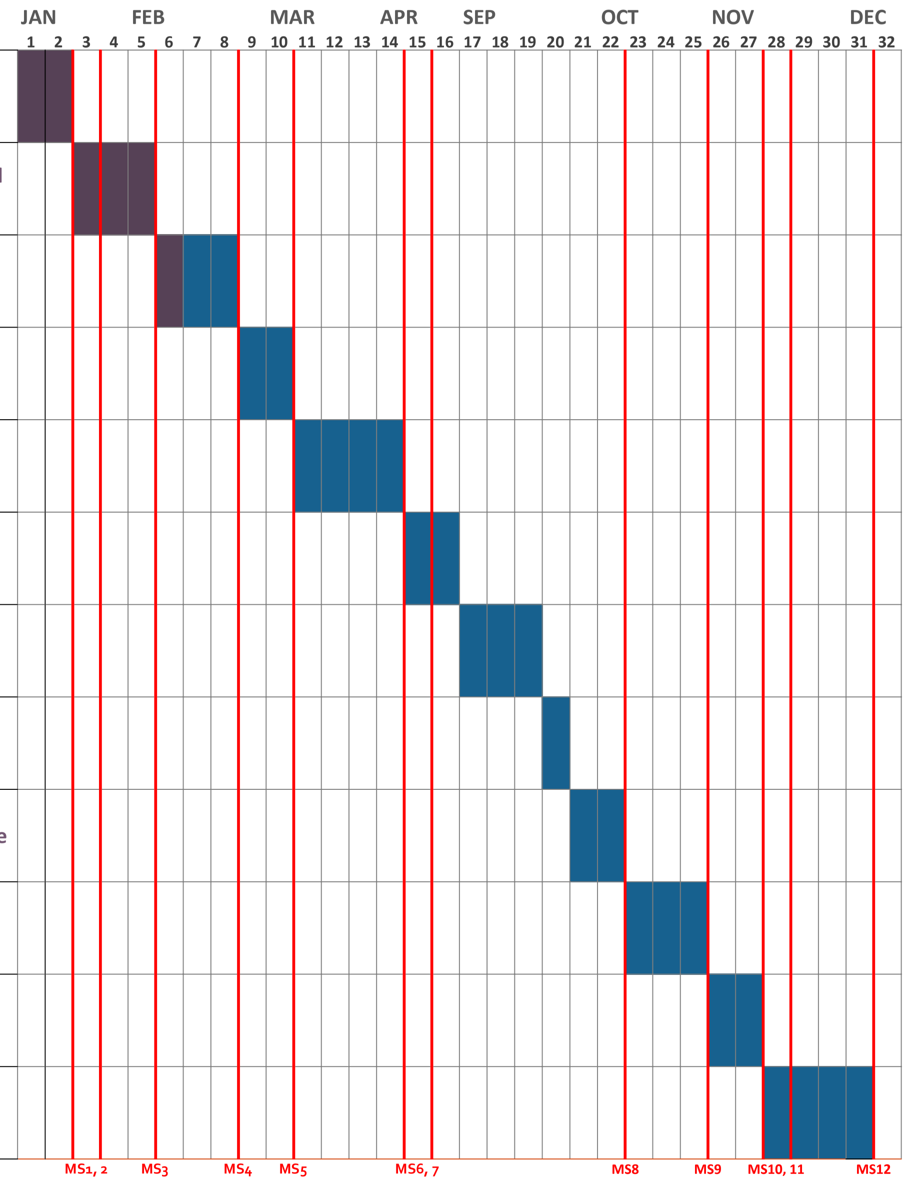
Appendix A: Project Management Plan for Year & Milestones

The Gantt Chart (Figure 1 on the following page) shows the team's project plan. There are a total of 12 phases and 12 milestones. The purple and blue cells represent the completed and estimated time for phases respectively, while the red lines represent the deadlines for the milestones. However, due to the possibility of unforeseen delays, these deadlines may not necessarily reflect the start or completion of their associated tasks. Each phase consists of a set of tasks that the team will work on. For example, the "finish the demo app" phase includes the combination of two apps: one developed by the backend team focusing on databases, another developed by the frontend team focusing on the User Interface. For "buffer time", the last week is left empty for any delays or changes at any of the phases. The hours of development depend on the difficulty and length of the phase.

MotorCity Casino

Current Wk 6 | Milestone | Completed Work | Estimated Time

Phase	Hours of Development	Phase Ending Milestone
Understand the Project/Planning / Build Team Cohesion	60.5	1. Initial Planning Complete
Literature Review / Infrastructure: Azure Devops, MS SQL Server, Android Studio	229	2. Purchase Beacons and Android Phones
Executive Summary 1 / Requirements Definition / System Diagram / Wireframe	239	3. ES1
Design Review 1/SPRING BREAK	162	4.DR1
Finish the Demo app	317	5. Demo app Complete
Design Review 2/ Executive Summary 2 / End of Term Wrap	162	6. DR2
Functionality + Initial user testing	242	7. ES2
Design Review 3 Report	86	8. DR3
Field User Testing /Validation Testing / Load Testing	165	9. Final Testing Complete
Final Version Cleaning	240	10. Final Version Clean
Contingency time	163	11. Sponsor Approval
Finalize Deliverable / Practice Final Presentation	321.5	12. Project Delivered



Weekly Hours of Development 77

Figure 1: MotorCity Casino project timeline

Appendix B: Literature & Technology Review

Introduction

Indoor localization, as its name suggests, refers to the task of determining the location of a user in an indoor space using a coordinate system defined for said space (Chawathe, 2009). However, the internationally standard Global Positioning System (GPS) cannot be used in indoor places since it cannot provide accurate location due to building roofs and walls obstructing the signal. Many different strategies have been proposed to achieve indoor localization such as the use of Wi-Fi signals or radio frequencies as most modern electronic devices have these capabilities. However, these methods were not necessarily practical due to the presence of many “blind spots” and the expenses associated with installing such hardware (Sharma, 2019). Bluetooth Low Energy (BLE) beacons, with their low cost, ease of deployment and low profiles, are an ideal technology for indoor localization. BLE-based localization is typically performed by installing a set of proximity beacons at known locations. Receivers will measure the received signal strength from the nearest beacons and use these values to predict their own position (Cannizaro et al. 2020). Then, based on the user's desired destination, a wayfinding algorithm will calculate a path for the users and display it on the app. This literature review from the hardware and backend team aims to analyze how BLE beacons can be used to calculate distances, compare popular localization algorithms in the Casino setting, and discuss efficient wayfinding methods.

Aside from the technical aspects, the user experience of our app is also of utmost concern. Since its development in the mid-2000s, mobile software poses challenges to the area of Human-Computer Interaction because of its differences with computers and keyboards (Punchoojit, 2017). The portable nature of mobile devices also provides the possibility of performing other tasks while using the phone such as walking (Harrison, 2013). Because of how different mobile devices are from traditional computers, the definition of good usability of software will be examined specifically on mobile devices. As a sub-category of mobile apps, the map interface has become ubiquitous in modern life and is thus a paramount part of the team’s app. In order to generate a satisfying experience for the users, different techniques to implement a user-friendly map interface such as overview+detail, icons, and font and color choices will be analyzed.

The Components of BLE

Bluetooth beacons have no knowledge of their location and are only capable of transmitting a signal that can be picked up by the devices around them. The behavior of beacons will depend on their configuration and transmit their UUID, major and minor values, battery level, current temperature, Adv count, their namespace, and instance IDs. Translating the transmitted information into a physical location requires the data to be processed, either on the device receiving the signals or on a server.

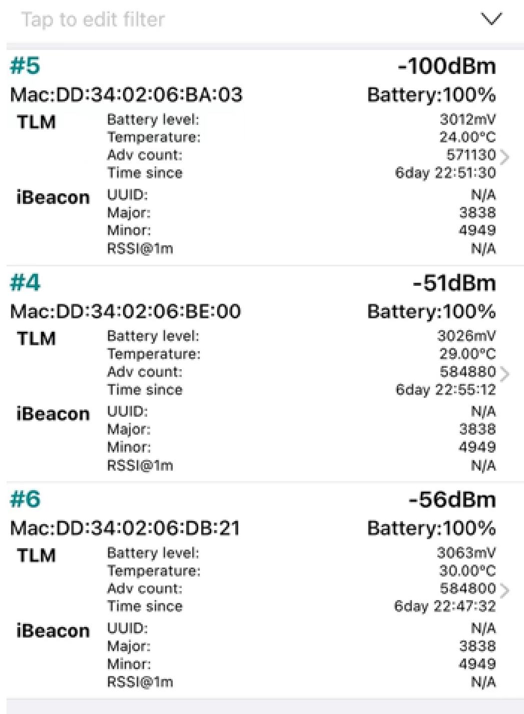


Figure 2: A screenshot from an existing Android application (KBeacon) showing the types of data transmitted by the beacons.

Building the System

The navigation system will be built based on a plan of the Casino floor provided by MotorCity Casino. The beacons embedded in the slot machines, along with any additional necessary beacons will be marked as “points of interest” (POI) to be stored on a database. A server to access this database of POI’s, determine a user’s location, calculate paths between points and transmit all of the data to a user’s device will also be created.

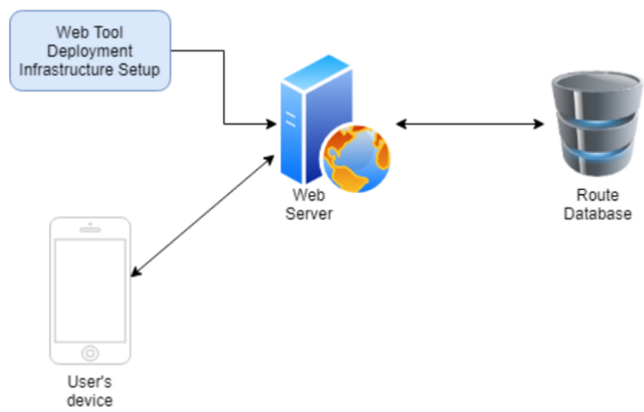


Figure 3: (*"BEACON DEPLOYMENT GUIDE: A STUDY ON BLUETOOTH LOW ENERGY BEACON INFRASTRUCTURE SETUP FOR INDOOR WAY-FINDING"*): Building blocks of a typical indoor wayfinding system.

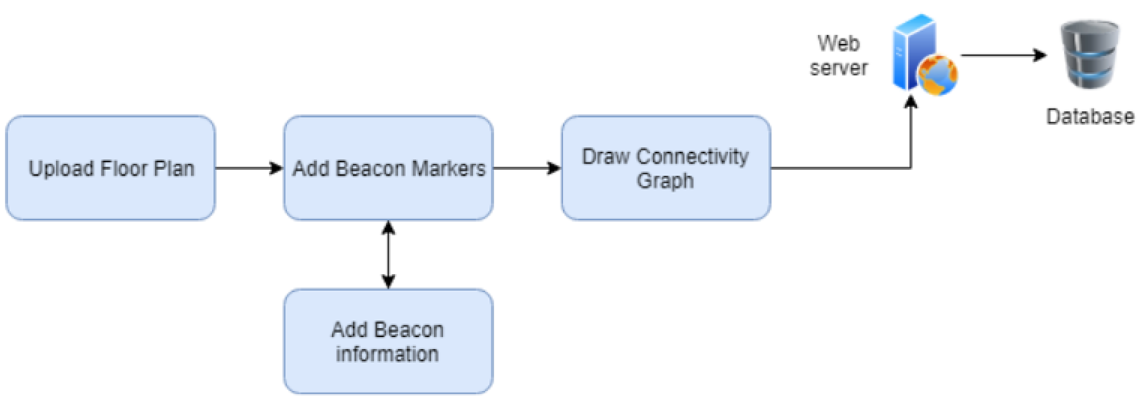


Figure 4: (*"BEACON DEPLOYMENT GUIDE: A STUDY ON BLUETOOTH LOW ENERGY BEACON INFRASTRUCTURE SETUP FOR INDOOR WAY-FINDING"*): The sequence of activities that can be performed on the web-tool.

Tuning the BLE Beacons

Because of the deployment of the beacons in close proximity, there is a possibility that too many beacons could be detected by a user's device - thereby potentially returning erroneous data. The Bluetooth beacons will have their Tx powers (signal strength) set to a level appropriate for the Casino floor to minimize this issue. For example, the signal of a beacon with Tx power 0dBm can be received by devices within a ~50m (164') radius in the absence of obstacles. To limit the maximum area of the place signal, the Tx power can be set to -16dBm so that the beacon can only transmit its signal in the range of 10m (in the absence of obstacles). This configuration process is expected to be highly time-consuming and require multiple measurements and calculations.

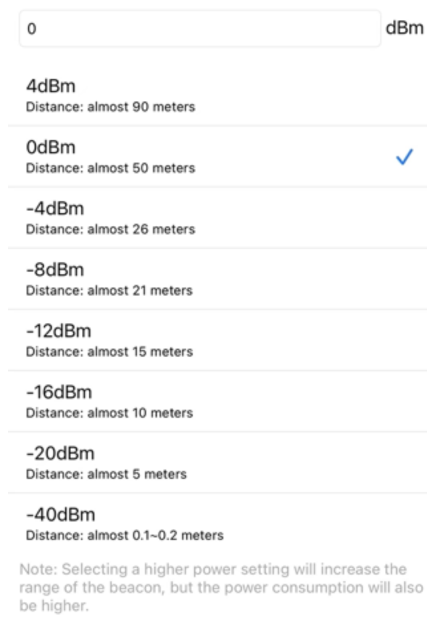


Figure 5: Different RSSI (Received Signal Strength Indication) at predefined distances.

Beacon Graph and Measurement

Once an appropriate Tx Power has been selected for the beacons, determining the location of a user is expected to be straightforward. As Chawathe (2009) states: "we model the range of a beacon as the set of locations from which it is detectable, without imposing any additional constraints, we can indirectly, but accurately, model the often complex artifacts of Bluetooth signal propagation due to walls, furniture, ductwork, etc". In MotorCity Casino's case, the signal from a beacon on one floor could be received by a device on the other. Thus, it may be necessary to model the range of a beacon signal as a sphere, as opposed to a 2-dimensional circle.

Localization Algorithm

In order to identify the user's current position on the map, a localization algorithm will be implemented using the received signal strength indication (RSSI) from the beacons. Cannizaro et al. (2020) explains that BLE-based localization algorithms can generally be divided into two categories: distance-based and fingerprinting-based. The distance-based algorithms will directly translate RSSI values into position coordinates for the object being localized without any preliminary measurements; however, at least three available nearby beacons are required for the procedure. In contrast, fingerprinting-based localization uses a vector of RSSI measurements in known fingerprint positions to create a "reference fingerprint map" (RFM) (Cannizaro et al. 2020). A machine-learning regressor is fed the RFM data to build an association rule between new RSSI measurements and their corresponding position estimates (Kriz et al. 2016, cited in Cannizaro et al. 2020). In an effort to determine the most suitable localization algorithm for this project, one popular algorithm from each category will be examined and analyzed in the Casino setting.

The comparative analysis conducted by Orujov et al. in 2016 discussed several distance-based algorithms including Proximity Localization, Centroid Localization, Weighted Centroid Localization, and Trilateration Localization. Orujov et al. set up a specific environment of 4.64m by 4.64m to test the algorithms, and the visual result shows good performance from both Weighted Centroid and Tilteration. While in Orujov et al. 's experiment, the weighted centroid has a slightly lower average error (0.97) compared to Tilteration (1.01): it is worth noting that there are four available beacons for the experiment. Since the accuracy of the weighted centroid localization depends on the number of signals the device can detect, the trilateration algorithm is preferable for this project as it only requires three beacons - lowering the cost of implementation.

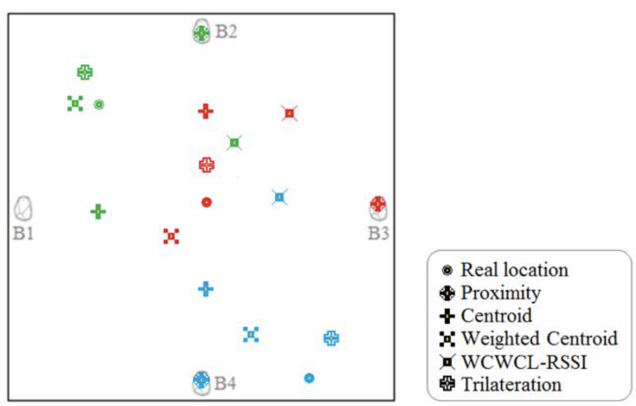


Figure 6 (Orujov et al., 2016): The visual result of distance-based localization.

In trilateration localization, the distance estimated by signal strength is presented as a circle with a radius defined by the beacon's RSSI on the device. Adding data from a second beacon reduces the plausible area to the overlap region of two circles, and the data from a third beacon will further reduce the region. Multilateration refers to the situation when there are more than three available beacons nearby. In the real-world Casino setting, the radii of these three circles may not always intersect at a point. For example, there may be locations on the map that are not covered by three beacons (no intersections of the circles) or where the exact location cannot be determined by the algorithms (the circle intersect at multiple points). Field testing will be required to identify these “dead zones”.

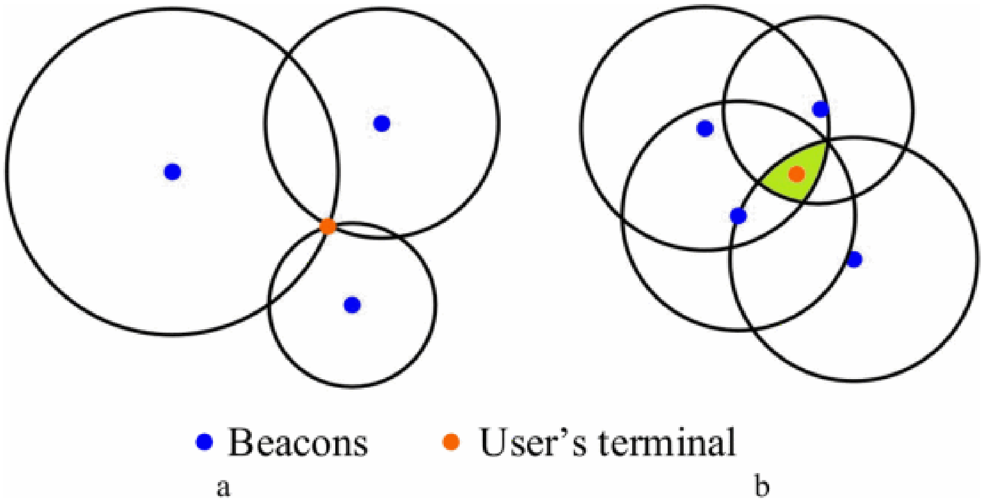


Figure 7: Trilateration and multilateration diagrams (a - trilateration, b - multilateration). Gorovyi, 2017

The detailed coordinates can be found solving the system of equations, shown in Figure 7, where unknown variables x, y are the coordinates of the user, $x_1, x_2, x_3, y_1, y_2, y_3$ are the known coordinates of the beacons stored in the database and d_1, d_2, d_3 are the estimated distances between the beacon and mobile device calculated using the RSSI.

$$d_1^2 = (x - x_1)^2 + (y - y_1)^2$$
$$d_2^2 = (x - x_2)^2 + (y - y_2)^2$$
$$d_3^2 = (x - x_3)^2 + (y - y_3)^2$$

Figure 8 (Orujov et al., 2016): System of equations for 2D Trilateration Localization.

As previously stated, it is worth noting that the Casino has two floors, and a three dimensional coordinate system may be required to indicate the user’s location. In that case, four available beacons will be required for the calculation. Furthermore, if more than three beacons are available, this system of equations can be rewritten in matrix form.

On the other hand, the k-Nearest Neighbors algorithm (k-NN) is a non-parametric supervised learning method that can be used for both regression and classification. When k-NN is used for regression, the output of the algorithm is the position of the object calculated as a weighted average of the values of its k nearest neighbors, where k is the main hyper-parameter of the algorithm. K-NN will store just the RSSI values and corresponding coordinates for the training data. At inference time, the k nearest neighbors of a new sample are simply identified evaluating a distance metric between its RSSI measurements and those of all training data (Cannizaro et al. 2020). When comparing the k-NN method with other fingerprinting algorithms including multi-layer perceptron (MLP) and support-vector machine (SVM), Cannizaro et al. evaluated them based on the training complexity and ease of adaptability instead of localization accuracy because all fingerprinting algorithms show similar error. This makes k-NN methods stand out due to its much simpler set of hyper-parameters (consisting solely of the number of neighbors k).

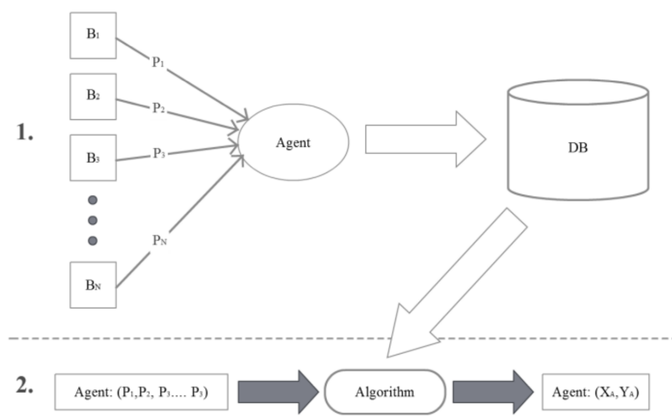


Figure 9 (Orujov et al., 2016): The scheme of the Fingerprinting Localization algorithm.

The differences between the two methods lead to strengths and weaknesses. Fingerprinting will require an additional configuration phase to store signal strength information from all beacons for each possible reference point. Although the fingerprinting method achieves great accuracy with a detailed database, the significant time-commitment in the data collection process must be considered. The trilateration technique is more flexible as the system will calculate the user's location in real-time and is more adaptable to environmental change than fingerprinting. However, Cannizaro et al. pointed out that trilateration is influenced by multipaths (radio signals reach the receiving antenna by two or more paths) and fading (attenuation rate of a signal due to obstacles) as they both affect the measured RSSI. For ease of implementation and flexibility, the project will start by testing trilateration localization and further adjustments (such as applying filters to reduce error, changing the beacon layout) will be made based on the test results from on-site visits.

Pathfinding Algorithm

After retrieving the user's current location, the app will show the user a path to their input destination. To calculate such a path, a pathfinding algorithm will be implemented. Pathfinding or pathing can be described as a computer algorithm that finds the shortest route/path between two points. The A-Star algorithm, or A*, attempts to calculate the shortest path / lowest cost route from the current/initial node to the destination node using one or more possible intermediate nodes. It is based on an evaluation function: $f(n) = g(n) + h(n)$, where $h(n)$ is an optimal path cost estimate from node n to the destination node, and $g(n)$ is the current cost from the current/initial node to any node n. As A* traverses the map, it follows the path with the lowest cost while keeping alternative nodes in a sorted priority queue. If a node being traversed has a higher cost than another encountered node at any point, it discards the node with the higher-cost and traverses the lower-cost node instead (Martinho et al. 2019). This process continues until the destination is reached.

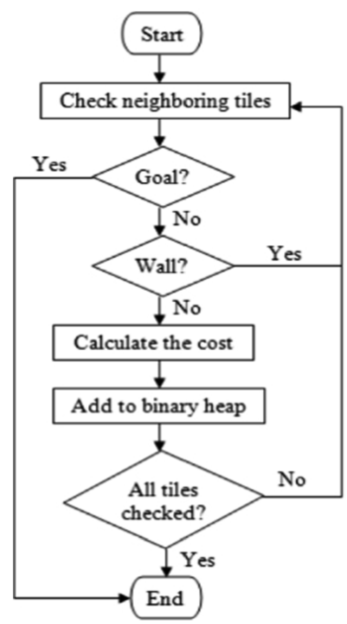


Figure 10: A* Search concept flow chart, Martinho et al. 2019.

In addition to the starting point and end point, the pathfinding algorithm also requires a map. In Cheraghi et al.'s model design (Figure 10), a points and lines table contains all essential graph data. A similar design will be adopted for the Casino navigation, where computation is done on the server. A challenge in the implementation will be navigation between two different floors.

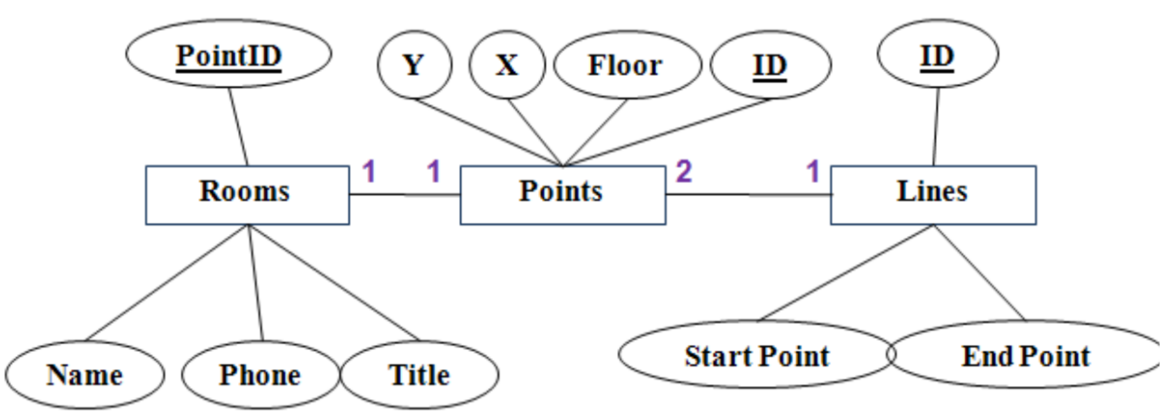


Figure 11 (Cheraghi et al., 2010): Database ER diagram for A* Search

Database Design and Application Programming Interface (API)

For the app to provide its users with correct information, a working database, that can parse beacon data, is required. Per MotorCity Casino's request, an Azure SQL Database will be used for this purpose.

The "Luch" beacon scanner library, created by Artem Gapchenko will be used. Luch was selected because it provides a robust API, support for many different types of beacons and is well documented on GitHub.

In addition to having a database that can read and parse the beacon data, this data must be sent to the end-users' device. In simple terms, an API allows two applications to communicate with each other. A REST API will be created with ASP.NET Core and that will allow the use of protocols such as GET, PUT, POST, and DELETE, which will be used to obtain or modify database data. The GET method is most applicable to this project because the user will often want to get information from the database that will be used to help determine their current location relative to their desired location. The API created by the MDP 2022 Cohort will query the database for a user request and return the relevant data. For example, if a user requests directions to the nearest restroom from their current location, the API can read this request and query the database on the MotorCity Casino server. The API will then return the user's current location and turn-by-turn directions to the restroom closest to them.

An alternative to the API is a web service that allows communication between two machines over a network. However, because a web service requires a network connection, an API is preferred for this project due to the relatively small scope - serving only the Casino floor.

Usability in Mobile Apps

Prior to the mid-2000s, the field of User Experience in Human-Computer Interaction was limited to computers and keyboards. As widespread adoption of touchscreen phones became common, many researchers in the field of Human-Computer Interaction began focusing on effective uses of User Interface (UI) elements to improve the usability of mobile touchscreens. To better understand how to construct intuitive UIs, the team analyzed the user experience in mobile applications in general and navigation apps in particular.

Punchoojit (2017) defines good usability in mobile apps as easy familiarization with the interface upon first contact and during later uses. Nielsen defines usability in terms of the following five properties: efficiency, satisfaction, learnability, memorability, and errors. (Nielsen, 1994, as cited in Harrison, 2013). Expanding on this, Harrison (2013) introduces three factors of the PACMAD (People At the Centre of Mobile Application Development) usability model: user, context, and task (Figure 11). The end-users of this app are customers who need to navigate through the Casino and the hotel staff who will deliver drinks to the customers. The context and task will be customers who need to find specific types of slot machines or other games. Thus, the team will focus on three aspects of Nielsen's five attributes, namely: user, task, and context.

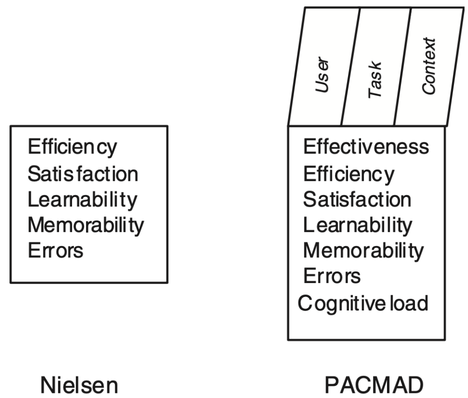


Figure 12: Comparison of usability models by Nielsen and PACMAD. Harrison, 2013.

To develop mobile software with good usability, the challenges of mobile software must be well understood. Both Punchoojit (2017) and Cockburn (2009) claim that one of the main challenges in mobile software is displaying information on a small screen. Harrison (2013), focusing more on the end-user side, stated that the limitation in sizes could lead to cognitive overload. Another stress on the cognitive load is that users are often performing other tasks such as walking when using phones. Due to this app's purpose, users are expected to be walking around the Casino floor. Therefore, efficiently displaying the hotel map and informing users of the necessary information (e.g. routes to slot machines) while they are walking is one of the greatest challenges for the User Interface/Frontend team.

To deal with the challenges the mobile devices pose, such as including multiple levels of details, ona small screen, must be kept in mind. Cockburn (2009) introduces several techniques including the Overview+detail display (Figure 12). Overview+detail would help give users a holistic view of the map on which they know their current location, while the detailed view that can be zoomed and pinched allows users to follow the route to their destination.

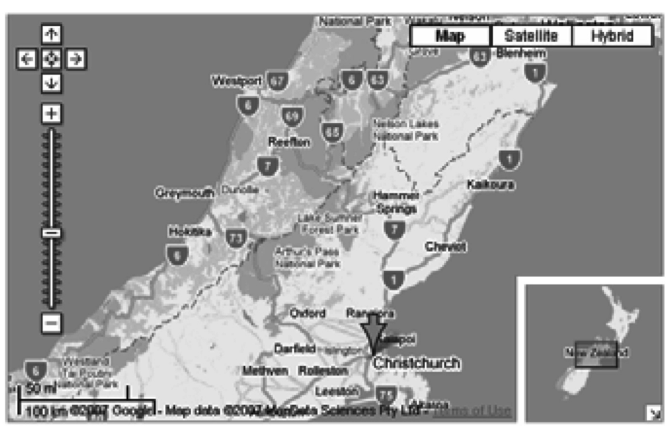


Figure 13: Google Maps. An example of Overview+Detail display. The bottom right corner shows the overview of the map while users can zoom the screen to view greater details. Cockburn, 2009.

Another technique that is important for our project is zooming. There are, as Cockburn (2009) claims, two types of zooming techniques: semantic zooming which means to display information in multiple scales (see figure 3), and portals that filter the information for easier understanding. Both types of zooming are crucial for the project: users need to use portals to filter out available slot machines and then zoom the map to see their way to them.

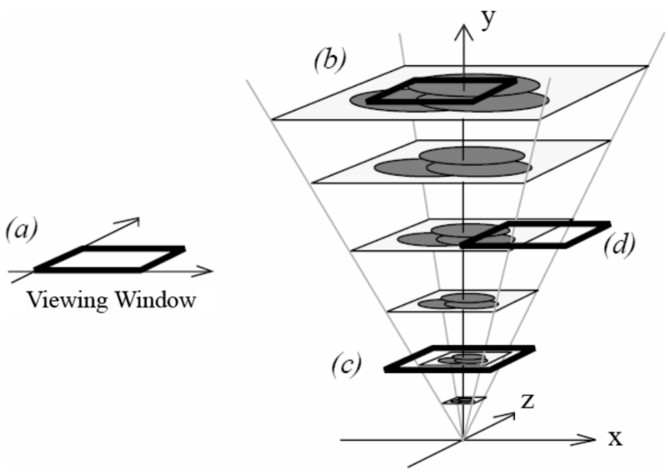


Figure 14: An image showing how to view objects in different scales by zooming. The viewing window could be our mobile screen. Cockburn, 2009.

Punchoojit (2017) also argues that some other challenges on small mobile devices include clicking on a navigation link. The app will be built using Webview, which displays a webpage in Android Studio. The navigation links will be styled as large buttons to make them easier to click on. Punchoojit (2017) emphasizes the importance of buttons as touchscreens remove the necessity of physical buttons. While this makes mobile screens larger, a lack of physical buttons results in no physical response and tactile feedback that decreases efficiency and user experience; adding virtual buttons can counter these negative effects.

Usability in Map Design

To deliver satisfying user experiences, the principles of cartography must be used to provide users with clear and easily understood directions on the limited space of the map interface. As Abuckley (2011) points out in her article. There are five main components for map design which are legibility, visual contrast, figure-ground, hierarchical organization and balance. The idea of legibility and visual contrast are the basis for seeing; visual contrast provides users with the ability to distinguish the map features and page elements from each other and legibility allows users to quickly decipher what they see on the map (Figure 14).

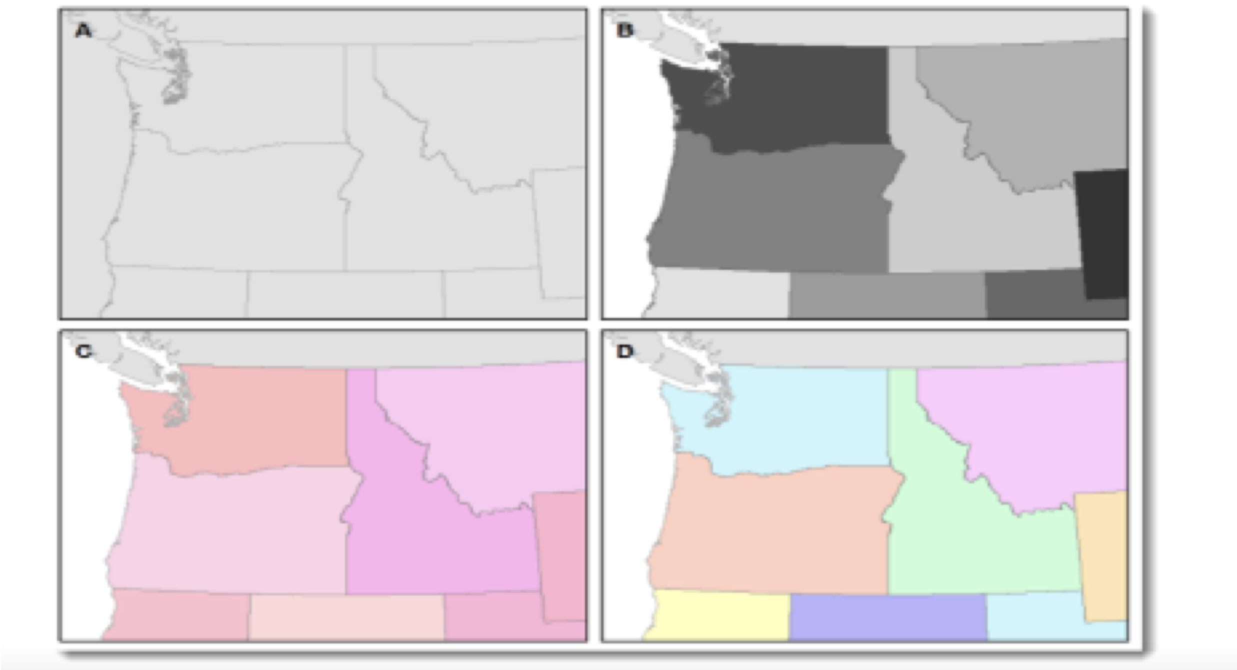


Figure 15 (Abuckley, 2011): The variation of colors in picture D can create a better visualization effect for the audiences.

Since each slot machine and game/service has relatively similar structures, it is crucial for the team to use colors and text to let customers easily track the machine they plan to access. In addition, although the Casino is not large, relative to many outdoor environments, since most customers will be unfamiliar with the floor plan of the Casino, it is easy for them to get lost in a relatively new environment. As a means of avoiding this potential problem, the third idea - figure-ground - will be employed. After the on-site visit of the Casino, the distinctive landmark for each part of the Casino and embed them into the map. The customers can easily track their current position without the help of beacons by finding one particular landmark they have seen from the map. This also helps customers to focus on a particular area of the map and can use the zoom in and out features without the distraction of the nearby.

Furthermore, the aesthetic aspects of the map design are extremely important. Robinson (1995) states that meaningful characteristics should be separated and portray likeness, differences and interrelationships. This is interpreted by the team as building maps with hierarchical structures and constructing layers of information. Some page elements will be with higher priority than other elements, and the importance of these should be emphasized by building visual planes. In practice, even similar icons or text have different meanings (Figures 15 and 16). Taking Google Maps as an example (2019), it can be observed that similar text or icons will have different meanings when indifferent fonts and colors. The most crucial elements should be highlighted using color, size or other method that immediately grabs attention. Furthermore, the technique of hierarchical structures comes into play when adding the search feature for the map interface. In modern navigation maps, several attributes are employed to rank each individual element appearing in the map to provide users with the most relevant information. Since the majority of the customers for the app will be looking for games/services in the Casino, these can always be the highest priority results.

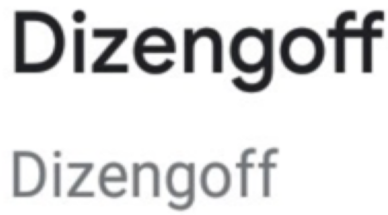


Figure 16 (SL,2019): The fronts in darker black will be used when displaying a certain location (more important) and the fonts in gray might be used to find a route from this location to other places (less important).

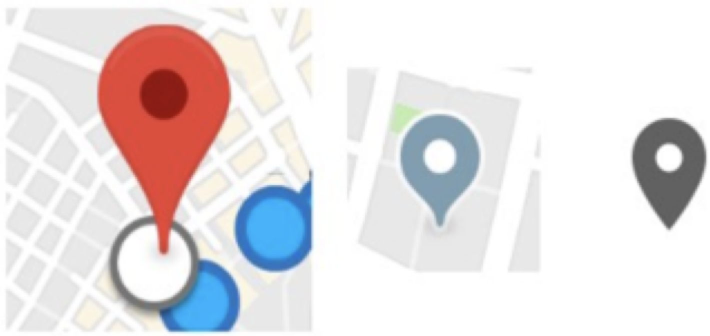


Figure 17 (SL,2019): Although they share the same icon. The icon in red is the highest priority since it points to the destination. The second icon can represent a place on the map and the third one will demonstrate a location off the map. The color shows the importance of each event.

As discussed earlier, displaying the map on a small screen can be challenging with current technology. Therefore, displaying the map "which results in an impression of equilibrium and harmony" (2011) can become a crucial task for us to provide the audience. In practice, since the map interface requires the constant use of zooming and pinching, the map cannot be guaranteed to maintain the effect of the expected design at all times. Therefore, the interface must be checked on a regular basis during the design process to ensure every location has obeyed the rules of harmony.

Conclusion

Combining the research done by the hardware and backend sub teams, the team has established how BLE beacons work and developed a web server profile that links a user's Android device with a database. However, because a site visit has yet to be conducted, the behavior of the beacons embedded in the slot machines must still be observed. The signal transmitted from the beacon is spherical and is greatly weakened when blocked by obstacles; thus the behavior of the beacons across floors must also be examined. Experiments with prototype beacons will be conducted during future visits to the Casino to address these issues further.

Challenges for designing mobile software that requires future research are limited connectivity and potentially high power consumption rate of the navigation app, as stated in Harrison (2013). Because the app is currently a prototype, there is insufficient information to investigate the power consumption and connectivity. One potential method to minimize power consumption that is being planned is to deal with all calculations and database accesses on the server-side instead of on the user's device. The current database design will store the beacons' location data, graph properties of the Casino map, and users' input. More parameters will be introduced in the future if necessary.

Bibliography

Buckley, A. (2011, October 28). *Design principles for cartography*. ArcGIS Blog. Retrieved February 6, 2022, from https://www.esri.com/arcgis-blog/products/product/mapping/design-principles-for-cartography/

Cannizzaro, D., Zafiri, M., Pagliari, D. J., Patti, E., Macii, E., Poncino, M., & Acquaviva, A. (2020). A comparison analysis of ble-based algorithms for localization in industrial environments. *Electronics (Switzerland), 9*(1) doi:10.3390/electronics9010044

Canterbury, A. C. U. of, Cockburn, A., Canterbury, U. of, Maryland, A. K. U. of, Karlson, A., Maryland, U. of, Benjamin B. Bederson University of Maryland, Bederson, B. B., & Metrics, O. M. V. A. (2009, January 1). *A review of overview+detail, zooming, and focus+context interfaces*. ACM Computing Surveys (CSUR). Retrieved January 29, 2022, from https://dl.acm.org/doi/abs/10.1145/1456650.1456652

Cheraghi, F., & Rajabi, M. A. (2010). A bluetooth-based indoor LBS system. Paper presented at the *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives, , 38*(4W13) Retrieved from www.scopus.com

Gapchenko, A. (2020, July 3). *Introducing Luch - a new library for BLE Beacon scans on Android*. Medium. Retrieved January 31, 2022, from https://proandroiddev.com/introducing-luch-a-new-library-for-ble-beacon-scans-on-android-50c1cb1623c

Gorovyi, I., Roenko, A., Pitertsev, A., Chervonyak, I., & Vovk, V. (2017). Real-time system for indoor user localization and navigation using bluetooth beacons. Paper presented at the *2017 IEEE 1st Ukraine Conference on Electrical and Computer Engineering, UKRCON 2017 - Proceedings,* 1025-1030. doi:10.1109/UKRCON.2017.8100406 Retrieved from www.scopus.com

Harrison, R., Flood, D., & Duce, D. (2013, May 7). *Usability of Mobile applications: Literature review and rationale for a new usability model - journal of interaction science*. SpringerOpen. Retrieved January 29, 2022, from https://journalofinteractionscience.springeropen.com/articles/10.1186/2194-0827-1-1

Martinho, S., Ferreira, J., & Resende, R. (2019). *Find_Me: IoT indoor guidance system* doi:10.1007/978-3-030-01746-0_27 Retrieved from www.scopus.com

Opeyemi, S. (2021, November 4). *Web API vs web service: Key differences, pros and cons*. Scraping Robot. Retrieved January 31, 2022, from https://scrapingrobot.com/blog/web-api-vs-web-service/

Orujov, F., & Maskeliunas, R. (2016). Comparative analysis of the indoor positioning algorithms using bluetooth low energy beacons. Paper presented at the *CEUR Workshop Proceedings, , 1712* 53-57. Retrieved from www.scopus.com

Punchoojit, L., & Hongwarittorrn, N. (2017, November 9). *Usability studies on Mobile User Interface Design Patterns: A Systematic Literature Review*. Advances in Human-Computer Interaction. Retrieved January 28, 2022, from https://www.hindawi.com/journals/ahci/2017/6787504/

Robinson, A. H., Morrison, J. L., Muehrcke, P. C., Passmore, L., & Starr, E. (1995). *Elements of cartography*. John Wiley & Sons.

Sharma, A. (2019). *Beacon Deployment Guide: A Study On Bluetooth Low Energy Beacon Infrastructure Setup For Indoor Way-Finding* (thesis). Retrieved January 29, 2022, from https://soar.wichita.edu/bitstream/handle/10057/16561/t19043_Sharma.pdf?isAllowed=y&sequence=1.

SL, A. (2019, March 5). *5 things I learned from recreating a google maps screenshot*. Medium. Retrieved February 6, 2022, from https://uxdesign.cc/5-things-i-learned-from-recreating-a-google-maps-screenshot-d34bdd83bd0a

S. S. Chawathe, "Low-latency indoor localization using bluetooth beacons," 2009 12th International IEEE Conference on Intelligent Transportation Systems, 2009, pp. 1-7, doi: 10.1109/ITSC.2009.5309711.