Implementation and Analysis of a Smart Team Management System using iOS Devices as iBeacon

https://doi.org/10.3991/ijim.v14i03.11680

Sindhumol S.
Cochin University of Science and Technology, Kochi, India
sindhumol09@gmail.com

Abstract—A cost-effective smart team management system using iOS devices as iBeacon for better task monitoring and presence detection is proposed in this work. It is designed to maximum utilize the proximity information offered by iOS devices through iBeacon integration. In addition to this positioning service, iOS core location framework, and push notification features are included to create a productive environment in an organization. Along with the employee tracking and task monitoring features, the proposed application provides the typical project management services like task allocation and notification, instant chat, status report, and employee log. On entering the iBeacon range, daily task details and announcements from all associated teams will be available in team member devices. The system is connected to a server for productive team management, and peer to peer communication is utilized for instant exchange of information. An experimental evaluation to study the effect of interference, obstacles, and measured power on received signal strength and accuracy of the proposed iBeacon is also conducted. The observed results positively recommend the proposed app as an efficient iBeacon application for team management.

Keywords—Team management, iBeacon, Core Location, iOS Application

1 Introduction

As technology grows faster, organizations and offices are also on the path of digital transformation, utilizing technologies like social media, mobile apps, big data, cloud computing, Internet of Things (IoT), etc. [1] [2]. But this digital revolution raises challenges in leadership, marketing management, process management, and project management. Team leaders and group leaders are struggling hard to track the activities of their team, or they often fail to keep the project deadline and budget. A recent survey report on project performance from 2011 to 2018 [3] says that nearly half of the projects are not completed on time owing to the poor task management and collaborative system. According to [4], task management apps can boost productivity through a collaborative work management culture among team leads and members in the office. Standard task management software features like daily reports, alerts, ethical employee tracking, and work monitoring has a significant role in achieving this. The success and reliability of the software depend on the technology used in application development. The widely accepted outdoor localization technology, Global positioning system (GPS) and geofencing [5] are proved to be very useful in task management applications. But they failed to be helpful in indoor environments for micro-location management. RFID and other technologies were also used in location management and team coordination [6], but the requirement of additional hardware devices and accessibility issues affected its popularity. In the last five years, Bluetooth Low Energy (BLE) beacons [7] replaced many of these existing technologies, with some exciting applications like indoor positioning, proximity detection, activity monitoring, etc. Beacon market could please the customers and users with a perfect blend of a digital and business user experience [8]. BLE beacons were introduced as a cost-effective method for indoor location management that provides low production cost, easy deployment, and better accessibility. The main advantage of the BLE beacon is, it can be attached to nonstationary objects also because it is estimating only the proximity (relative distance to an object), not the exact location. A detailed literature review on BLE applications in IoT is available in [9]. In 2013, Apple Inc. introduced the BLE protocol standard, iBeacon, to communicate with iOS devices [10]. Since then, other vendors also entered the market with iBeacon compatible devices. iBeacons are now compatible with Mac machines (with OSX Mavericks and later), iPad mini or later, iPhone 4s or later, android 4.3+, and windows phone devices. Also, there is an option to turn the iPhones, iPads or BLE compatible devices to an iBeacon through iOS Bluetooth Framework and Core Location framework [11].

In this work, the facilities provided by iBeacon, iOS location mapping, geofencing, and push notification are best utilized to propose an iOS application for smart team management. Compared to the existing team management applications, the proposed app completely avoids the installation of stationary physical beacons by turning team members' iPhone or iPad into iBeacons. At the same time, it integrates all the facilities provided by the physical beacon with the team management features in a single iOS application. It is implemented particularly for one team, where the team leader and employees have separate login access and ID for accessing the tasks. Once the connection with the login server is established, the team leader can view the active members, add more employees, set the task, view the status, chat, and see the employee log and so on from the dashboard. The complete system is connected to a server to provide better communication among the team members. Peer to peer communication through push notifications and chat also added to the system. A team member is allowed to view the assigned tasks, notifications, and active colleague list. The proposed app provides an employee log that summarizes the employee entry and exit in the beacon range as well as his performance in the task completion. It also facilitates a smart employee tracking through automatic presence detection, thereby avoiding long queues for swiping or punching. In this work, both user and team lead can enable iBeacon broadcasting, and with the help of push notifications, one can wake another inactive app. A feasibility study on turning iOS devices into iBeacon for team management is also conducted. The performance of the proposed app on varying beacon ranges and different measured power was observed. Effect of interference from other beacons in proximity variation, and received signal strength variation in the presence of different obstructing objects were added in the experimental study. Results of the exploratory analysis and software testing positively recommend the proposed system as a reliable and acceptable application for typical office environment.

The remaining sections of this paper are organized as follows. Section 2 discusses the recent related works in beacon proximity applications and project management. Section 3 gives an overview of iBeacon architecture and implementation details of iOS devices integration as iBeacon. Proposed app design and architecture are also added to this section. Section 4 explains the app implementation and screenshots. The experimental results of the proposed iBeacon are analyzed and discussed in Section 5. Section 6 concludes the paper with an insight into future work.

2 Related Works

BLE beacons, with its proximity services, confirmed its position in almost all domains, including retail, finance, healthcare, education, automotive, etc. In retail applications, BLE beacons in a store are connected with the shopping app installed in customer phones. Retailers correctly identify the location of the customers and send them relevant information and promotions [5]. The main challenge in this implementation is turning Bluetooth on and making their app active [12]. Apart from retail applications, banks [12], airports [13], hospitals [14], hotels, and companies are now extensively using these beacons for different services like employee tracking, cab services, conference alerts, etc. iBeacon and BLE beacons are found to be useful in learning and teaching also [15][16]. BeaLib is a smart app [17] providing quick and easy access to users having smartphones. Without using an internet connection, the library user gets the necessary information on the mobile app. In [18], an in-room presence detection system is introduced to automate the attendance collection of conferences or courses. There are several works related to smart office management using mobile apps available in recent works [4][19]. Asana, Microsoft planner, Trello, Wrike, Smartsheet are a few popular apps [4] among them. The beacon specific features in indoor positioning made it acceptable in app-powered office management systems also. SOEMS [20], is a BLE beacon-based system with a mobile app to reduce energy consumption in the office without causing user inconvenience. The power-saving mode of the user's PC, lights, and fans are automatically controlled on user entry and exit. Beacons in an office can ease the coordination of group work or teamwork with the support of smartphones and apps. In AllHours App [21] daily work schedules of employees are maintained by passing through a fixed micro-location known as "check-in" location where beacon is installed. StaffNearBy[22] is another application using Beacons for attendance management, employee tracking, crew safety management, and location-based information sharing. DE-TASK [23] is an android app for task management using Firebase. It provides task management features like user creation, location tracking, beacon management, and task management. To work with DE-TASK, admin mounts the beacons in office premises for different departments, and assign the tasks to the user after selecting beacon. A lot of employee tracking and task management apps are available in App Store and Google play store. But they are not completely dedicated to integrated team management. Either they provide some proximity related services, or maybe they are based on separately installed iBeacons. These limitations are well addressed in the proposed app through the implementation of an iBeacon integrated team management system.

3 iBeacon Architecture and iOS Integration

iBeacon technology was introduced by Apple Inc. in iOS 7[10], as a BLE based protocol enabling location awareness around an object (can be static or moving). iBeacon can be considered as a subset of BLE beacons, providing automatic and location-specific services using mobile apps [24]. iBeacon technology involves hardware and software parts [25]. Hardware device consists of a wireless transmitter and a smart device for BLE transmission. The software part, a smartphone app, is used to analyze the received message and invoke the relevant actions. iBeacon transmitter advertises a stream of messages at a regular interval of time, and the receiver detects it based on the proximity of the device [26]. In contrast to the android devices, iOS devices regularly search for the iBeacon signals, and the app becomes active when the device is in iBeacon proximity. Any smart device having a BLE receiver can receive iBeacon message from a separate iBeacon hardware, or iOS device turned into iBeacon [27]. The range of beacon transmission depends on location and obstructions in the beacon range. Standard beacons often show a range of 70-100meters, whereas long-range beacons transmit up to 300 meters.

3.1 iBeacon message structure

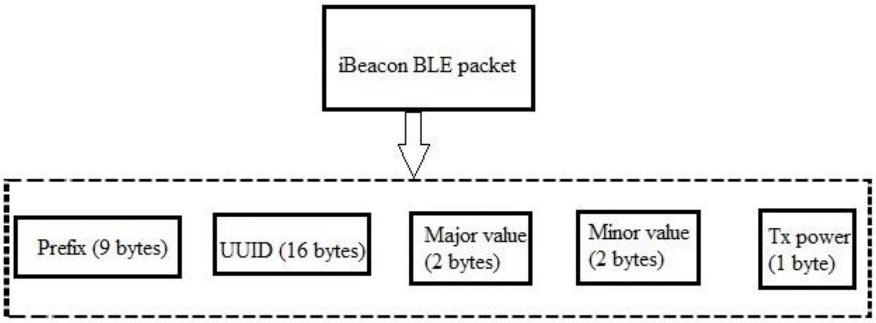


Fig. 1. iBeacon payload format

An iBeacon architecture, as shown in Figure 1, defines the information with the prefix, UUID, major and minor value, and transmission power (Tx)[10]. An iOS device can use a single UUID to represent the entire organization. The major number groups a related set of beacons and minor number distinguishes the individual beacons. Suppose an IT firm has offices in U.S, Japan, India, and each office have different teams like product team, development team, and testing team, iBeacon architecture can be configured with the same UUID for IT firm, different major values for different locations, and same minor value for a particular team as shown in Figure 2.

The dotted rectangle in Figure 2 shows that minor values are common to a particular team in all locations. For example, minor1 is assigned to the product team from all three offices. Perhaps the employee is associated with the development team in the US; the beacon configuration can be set as UUID+Major1+Minor1. This distribution makes it easy to distinguish the team beacons from one another. The proposed app follows the same distribution in its architecture, and it provides the option to decide the granularity to the organization or team, which can be configured at runtime. More details on UUID generation and iBeacon settings in this work are explained in section 3.3.

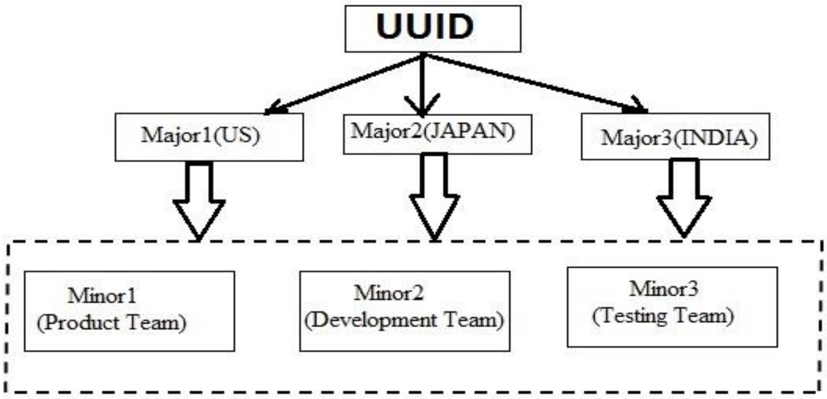


Fig. 2. iBeacon distribution among different teams

3.2 iBeacon performance measures

Received Signal Strength Indicator (RSSI), Positioning Accuracy, Proximity, MeasuredPower (Tx) are the measures used in the beacon performance analysis of this work. RSSI is defined as the received signal strength in decibels measured from one meter away from the iBeacon. The positioning with RSSI estimates the receiver position through the measurement of received BLE signal intensity [28], which can be varied according to the signal attenuation from the environment. Positioning accuracy is a measurement based on an estimated position compared to the actual position [26]. It can be varied according to the signal loss in the propagation path. The capacity of the receiving antenna, characteristics of the transmitter and the environment, interference from other wireless devices and beacons, and room layout can affect the received signal strength and thereby accuracy or proximity calculation. Several positioning accuracy estimation algorithms exist in the literature [25][26][28]. In this work, accuracy values are measured in meters from a variable returned from the iOS framework. Proximity values are categorized into ‘Immediate’, ‘Near’, ‘Far’, and ‘Unknown’ based on the received accuracy. It is only an indicator to show the relative position of the device around the beacon. Another measurement used in this work is the Measured Power or Transmit Power (Tx) as specified in the iBeacon message structure [24]. Measured power in iBeacon represents the signal strength of the beacon for a receiver at one meter away. Measured power is often set by the manufacturer as a pre-configuration parameter. However, as a part of calibration at the time of iBeacon installation, it can be adjusted to overcome any attenuation in a cabinet, or to compensate with the reflections against a metal frame or Wi-Fi interference. Range of Transmit Power values and settings vary in different beacons; a detailed explanation is available in [24]. It is different from the transmitted power, the capacity of the transmitting device. The article [29] explains how to choose the transmitted power to prevent the region overlapping. iBeacon uses the 2.4 GHz BLE to broadcast signals. So, signal loss due to physical objects, human body, and obstacles like walls, doors or other physical structures has to be considered at the time beacon installation. A detailed experimental analysis of all these cases is discussed in section 5.

3.3 iOS device integration to iBeacon

In proximity applications, usually dedicated iBeacon hardware is used along with mobile apps. According to Apple iOS documentation [11], any iOS devices that supports BLE can be turned into an iBeacon using the iOS frameworks like Core Location and Core Bluetooth. Three main steps exist as per documentation to change an iOS device into an iBeacon transmitter.

- **Step1**: Generate a 128-bit UUID for the device to uniquely identify an item or organization. In iOS, the NSUUID class is used to handle the UUID values. Terminal Mac app in Mac machines can be used to generate one UUID. In the Terminal window, type ‘uuidgen’ and press return. The generated UUID value will be a hexadecimal value of 4 sections, for example, ‘D2B5-45B9-86F0-765531FFA8B7’.
- **Step2**: Use the UUID value from Step1, and set a major and minor value as discussed in Section 3.1 to create a CLBeaconRegion object representing beacon ID.
- **Step3**: Configure the iOS device as a Bluetooth peripheral using CBPeripheralManager class from the Core Bluetooth framework, the code snippets are available in iOS documentation [11]. Start beacon data streaming to other devices in the short-range region. Here, Bluetooth hardware part in the device is utilized to broadcast the information.

In addition to UUID, major, and minor value, peripheral data provides a parameter ‘withMeasuredPower’ to represent Tx value. By default, it is set to ‘nil’, specifying default measured beacon strength from one meter away. iOS Core Location framework provides CLLocationManager class to work with beacons; startRangingBeacons method returns CLBeacon instance that identifies an array of beacons within the device range and startMonitoring method registers a beacon to the device for ranging. iBeacon details like UUID, Major, Minor, and performance measures like accuracy, proximity, and RSSI can be collected from CLBeacon instance in the iOS framework. For a particular app, there is no restriction for beacon ranging, but monitoring is restricted to maximum 20 regions.

The proposed app follows the same steps to turn the team device and member iPhone/iPad to iBeacon. After a successful login, the array of beacons in the device range is identified by the startRangingBeacons method. When the device is wakened, it pings the all registered beacon even if the app is closed and generates a notification to make the app active or open. In this work, these wakening up can be done in two ways; users and team leaders have the option to turn their device into iBeacon, and both ends can send push notifications. The proximity of the device can be categorized into:

- **Immediate** (< 1 meter) – showing the physical device in its close range.
- **Near** (1-3 meter) – If there are no obstacles, and the receiving device is set on a direct line of sight from the transmitter, 1 to 3 meters will be the range. Otherwise, the actual 1-3 meter will show different proximity. A detailed discussion is available from Section 5.
- **Far** (> 3meter) – In the ideal condition, it is higher than 3 meters, actual values may change according to the variation in RSSI.
- **Unknown** – This is the value shown, either before monitoring starts (initially) or when the beacon is not active or out of range.

3.4 The architectural design of the proposed app

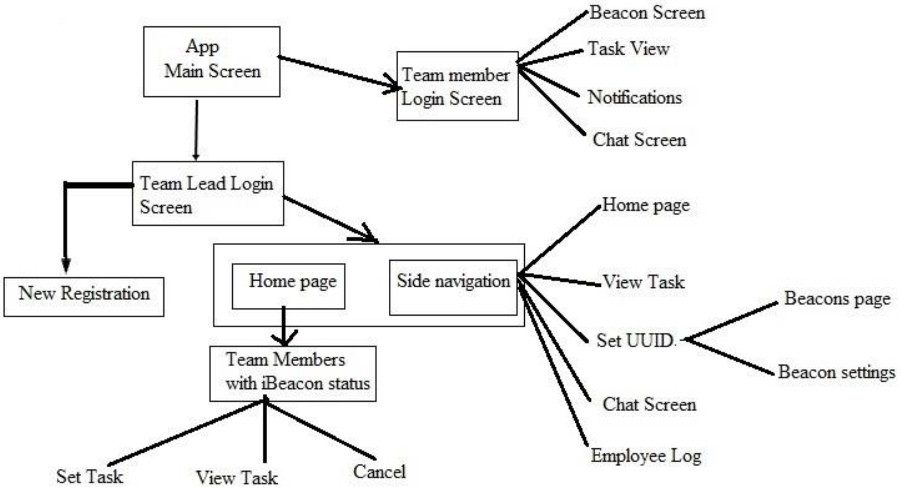


Fig. 3. The architectural design of the proposed app

The proposed team management app launches with an option to login as Team Lead or User (Employee) as shown in Figure 3. Successful Team Leader Login navigates to:

- View the active members
- Task View with status
- Set the new task by choosing employee individually or by group
- Chat with employee/Send notifications to members
- Set the TeamID/UUID
- View the employee log (with entry and exit time, task report)

- Log out option

User Login navigates to:

- View the active members on the home screen
- View assigned task from team leader/s
- Chat with team leader and colleagues
- Send a request to the team leader for acceptance/awake
- Log out option

The design phase of a mobile app starts with a prototype design, which outlines the different screens of the proposed app and its navigation from one screen to another. Also, it involves the design of the page layout, interface elements, and workflow. There are different methods of prototyping; the Wireframe model [30] is the basic one; tools like JustinMind[30] are also available for the prototype design. In this work, the app framework including screen prototypes and navigation as specified in Figure 3 were first designed with JustinMind before starting iOS implementation.

4 Implementation of Proposed App and Output Screens

4.1 Software and hardware components

The complete development and implementation of this app were performed using Macbook Pro with Intel Core i5 2.7GHz Processor, 8GB RAM, and macOS Mojave. XCode version 10.2.1 with Swift 5 [31] provided the software development environment. Screens and layouts were designed using Storyboard [31], and the business logic was implemented with Swift programming language and backend services. The database in this app used the Core Data framework and SQLite for the implementation of backend services. Communication with the server was implemented using AlamoFire [32], which provides a chain of request/response methods and JSON parameters. The security keys used in this work for accessing location and allowing notifications were stored in info.plist of the XCode project[31].

4.2 App screen implementation and screenshots

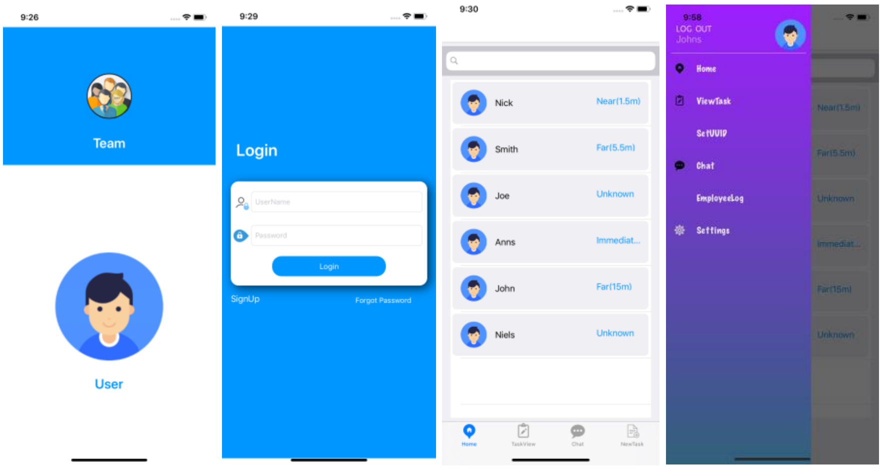
The relevant screens from the implemented app are shown in Figure 4(a) to Figure 4(h). In addition to these pages, screens like a chat window, new registration, settings, employee log, forgot password, logout, etc. also exist in the implementation. The app launches with the main screen as shown in Figure 4(a), where the user can choose between User or Team. Here a segue action [31] for both the user and team leader is created to invoke the corresponding login page.

Login screen: The login screen, Figure 4(b), provides three options; login, sign up, or forgot password. Successful login directs to the Home page. Entering the wrong credentials denies login to the app. ‘Forgot password’ button leads the user to a FORGOT

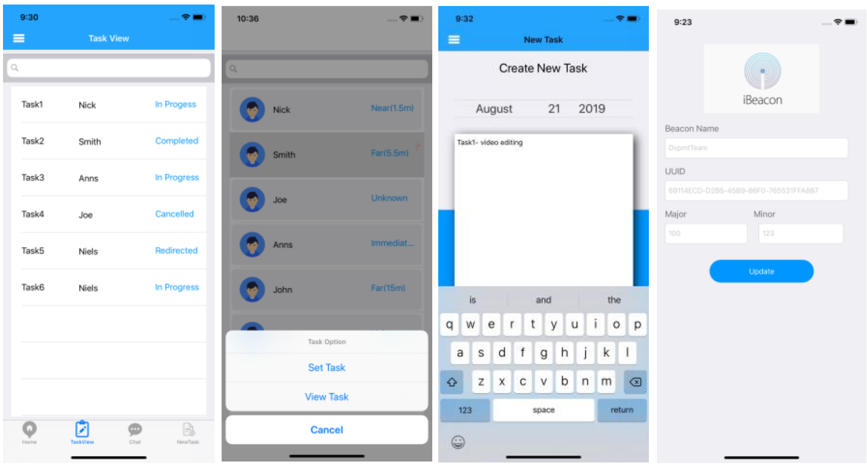
PASSWORD page, where he can give the email address to which the new password is to be sent. ‘Signup’ button leads to the new registration page.

Home screen: Home screen, shown in Figure 4(c), can be reached either by a successful login as a team lead or from the side navigation after login. It is the main screen of the proposed app where the current proximity status of the members is shown. The table view displays the team member list with dynamic cells having Image view, two Text fields, and a Search bar. The proximity status of each member is shown at the right end of the row.

Side navigation: Figure 4(d) shows menu items like Home, View Task, set UUID, Chat, Employee Log, Log out on the left edge of the screen. It is hidden most of the time, but it is revealed when the user swipes from the left of the screen or when the user touches the app icon at the top left.



(a) (b) (c) (d)



Task1	Nick	In Progess
Task2	Smith	Completed
Task3	Anns	In Progress
Task4	Joe	Cancelled
Task5	Niels	Redirected
Task6	Niels	In Progress

Nick	Near(1.5m)
Smith	Far(5.5m)
Joe	Unknown
Anns	Immediat...
John	Far(15m)

Create New Task

August 21 2019

Task1- video editing

Beacon Name: DvpmtTeam

UUID: 69114ECD-D2B5-4589-86F0-765531FFA8B7

Major: 100

Minor: 123

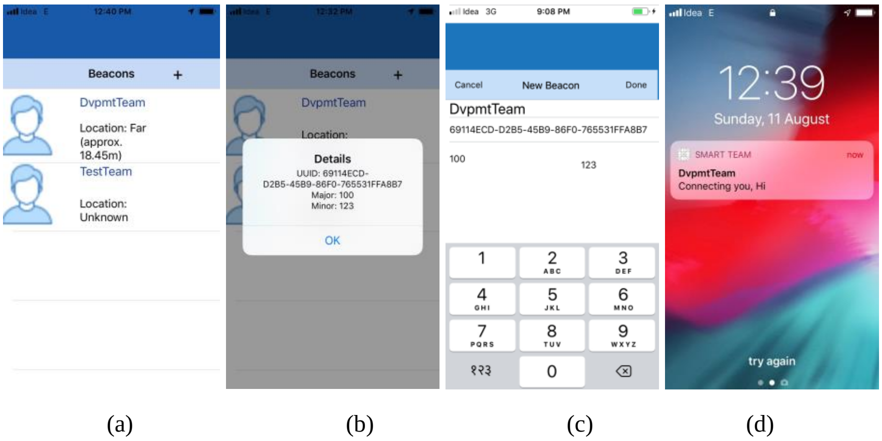
(e) (f) (g) (h)

Fig. 4. Output Screens: (a) Main Screen, (b) Login Screen, (c) Home page, (d) Side Navigation, (e) Task view, (f) Task management popup, (g) New Task Screen, (h) iBeacon Settings

Task view: As given in Figure 4(e), team tasks are listed out in a Table view with details like task description, assignee, and status (In progress, redirected, canceled or completed) in each row. The team lead can search for various tasks listed in the table view. Team members can be selected (as a group or individual) to assign a new job for them. The selection of a row shows a button(arrow) on the right end of the row to popup an action sheet to create a new task or manage the existing one, as given in Figure 4(f).

New task: This option, as shown in Figure 4(g), can be used to write and assign a new task with its deadline to a team member. While pressing ‘return’, the new task will be saved to the existing task list, and a notification will be sent to the selected user/users.

iBeacon settings: This is a unique page for the team lead, to set member ID as UUID with major/minor values to get connected with the team, see Figure 4(h). It follows the naming structure as described in section 3.1. Team Lead is in control of setting all these fields through this screen.



DvpmtTeam
Location: Far (approx. 18.45m)
TestTeam
Location: Unknown

DvpmtTeam
Location:
Details
UUID: 69114ECD-D2B5-45B9-86F0-765531FFA8B7
Major: 100
Minor: 123
OK

DvpmtTeam
69114ECD-D2B5-45B9-86F0-765531FFA8B7
100
123

DvpmtTeam
Connecting you, Hi

(a) (b) (c) (d)

Fig. 5. User side iBeacon list and notifications: (a) Registered iBeacon (b) iBeacon details (c) Adding new iBeacon (d) Notifications

As per the design of this app, users are allowed to connect with more than one team through iBeacon. Users can configure their iPhone/iPad using the ID assigned by the team lead for BLE signals transmission as an iBeacon. Also, the user can search for the ranging teams connected to him. To enable these features, the user should provide location access and notification permission to the app on activation. A successful login navigates to a beacon page displaying all the connected teams like DvpmtTeam and TestTeam as shown in Figure 5(a). Each row displays the team name, current proximity, and accuracy of the user relative to iBeacon. Double click on each row gives iBeacon parameters as shown in Figure 5(b) using an alert in iOS. The user is equipped with an option to add more teams with new beacon settings to the list, as displayed in Figure 5(c). Once a beacon is registered to the user app, all notifications on entry and exit of that beacon range are pushed to the user as shown in Figure 5(d), even though the app is not active or not in the foreground.

5 Analysis of iBeacon Accuracy in the Proposed App

Performance analysis of turning an iOS device into an iBeacon in different environments are discussed in this section. MacBook Pro with Intel Core i5 2.7GHz Processor, 8GB RAM and macOS Mojave, iPhone 6s, iPhone SE, and iPad mini4 were used in the testing and analysis. Observations were manually recorded; for each location, 10 consistent RSSI and distance measurements were taken at a time, and the average value was recorded. Four sets of experiments were conducted in the performance evaluation. The first one was the study on variation in accuracy values against the actual distance. The second experiment evaluated the effect of signal interference on proximity and accuracy. RSSI and Accuracy changes on varying MeasuredPower (Tx) were considered in the third set. The fourth set was a study on the effect of obstacles like cardboard, human body, etc. on accuracy variation. Details of each set of experiments are discussed in the following sections.

5.1 Effect of actual distance change on accuracy values

Experiments to study variation in accuracy values (here, measured in meter using iOS SDK) in an ideal situation as well as in an environment with obstructions are discussed in this section. Devices turned into iBeacon were placed at a table height of 80cm, facing the transmitter on a straight line with the receiving end. Actual distance varied from 1m to 20m to collect the experimental accuracy values. Table 1 summarizes the observed results.

Table 1. Accuracy variation against actual distance

Actual Distance (m)	Accuracy (m) (Without obstructions)	Accuracy (m) (with obstructions)
1	1.03	
2	1.93	
3	3.08	
4	3.92	
4.5	4.6	8.89
5	4.9	15.8
6	6.11	30
7	7.18	34
8	8.33	47
10	10.53	48
15	16.15	68
20	23.4	76.5

The first column represents the actual distance values, with a straight line of sight from iBeacon to receivers. Observed accuracy values without any objects on its propagation path are tabulated in the second column. The third column represents the values in a real environment with walls and doors, where signal strength changes could be observed only from 4.5 m to 20 m. From these results, it is observed that initially, there is not much difference exist between actual distance and measured accuracy. From an actual distance greater than 6m, accuracy values are found to be deviating from the expected value. For actual distance 4.5m, the difference was approximately 0.1m, whereas, for 20m, the gap increased to 3.4m. While considering the observed results with obstructions, it is found that accuracy values are multiple times the actual distance. A drastic increase in accuracy value observed on increasing actual distance. For the last two values, 15m and 20m, there was a wall and a wooden door on the line of sight while taking the measurements. Signal strength decrease and accuracy change due to obstacles, as discussed in section 3.2, are evident from the corresponding accuracy deviations, 53m, and 56.5m respectively. The observed results confirm that positional error (the difference between the expected value and obtained result) increases as the distance of the receiver from iBeacon increases.

5.2 Effect of interference on accuracy and proximity

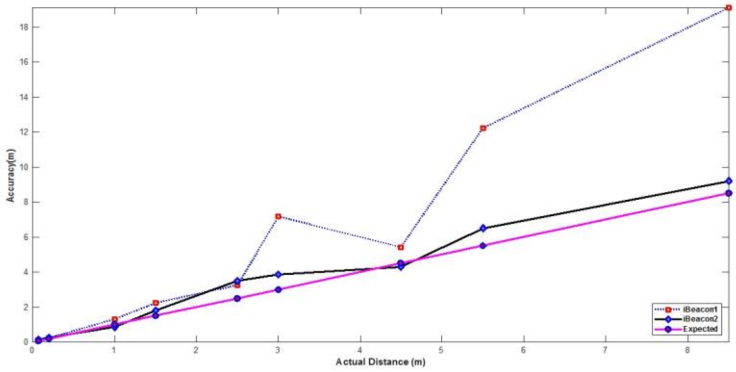


Fig. 6. iBeacon accuracy variation in the proximity of another iBeacon

The proposed app may generate interference on iBeacon transmission through a simultaneous activation from multiple users. The second set of experiments in this work is dedicated to a study on the variation in accuracy and proximity due to signal interference. The experimental values of these performance measures were collected in a real environment, having WI-FI, and the presence of other mobiles. One iPhone SE and iPhone 6s were placed at a table height of 85cm, without having any obstructions from walls or other objects on the line of sight. Accuracy values were manually recorded from the immediate neighborhood to 8.5m. Another iPhone 6S was used as the receiver. When the receiver was very close to both beacons, received accuracy values were almost same, as shown in Figure 6. For .5m to 1.5m, slight deviations from actual distance plotted for accuracy values from both beacons. After 2.5 m, accuracy values from iBeacon 2 found to be keeping a consistent variation from actual distance, whereas iBeacon1 accuracy values are highly fluctuated, especially after 4.5m. These fluctuating results show that signal interference from other devices and iBeacon has a significant role in adversely affecting the iBeacon performance, especially when distance from beacon increases in a real environment.

5.3 Effect of measured power variation on RSSI and proximity

As mentioned in section 3.2, Measured power value causes the receiver to change its proximity calculation so that accuracy ‘1m’ is measured when the receiving RSSI is same as the sent Measured power value. The variation in Measured power values in different experimental environments are explicitly described in this section. At the time of the beacon calibration, the receiver should be placed at ‘1m’ in a typical direction to receive the signal over 20-30 seconds. Consistent RSSI value will be considered as the Measured power to be set in the iBeacon device.

Table 2. Effect of Measured Power variation on proximity and accuracy

Measured Power(dB)	RSSI (dB)	Proximity (Accuracy in meter)
Default	-57	Near (1.03)
-120	-57	Immediate (0.17)
-110	-58	Immediate (0.19)
-100	-57	Immediate (.20)
-90	-58	Immediate (.22)
-80	-58	Immediate (.28)
-70	-59	Immediate (.32)
-60	**-59**	**Immediate (.99)**
-50	-58	Near (2.91)
-40	-58	Far (5.24)

Table 2 describes this calibration, using an actual distance of 1m. In this third set of experiments, 'MeasuredPower' changes from -120dB to -40dB at an interval of 10dB. Default value also recorded; as per iOS settings its value is 'nil' by default. All the measurements were taken in an ideal situation avoiding all interferences and obstacles. iPhone SE was set as iBeacon, and iPad mini 4 was the receiver. Experimental results show that accuracy 1.03m and proximity 'near' is recorded for default RSSI value, -57. It can be seen from Table 2 that when 'MeasuredPower' is set as -120, RSSI is same as the default value. But, proximity is shown as 'Immediate', with an accuracy value of 0.17m. The receiver identified the relative position as very close to the beacon. The same phenomenon repeated for all values from -110 to -60, with accuracy values gradually increasing. For Measured power -60(close to default value), it is found that accuracy is .99m (close to expected 1m). i.e., when the receiver has better RSSI than the Measured power, its proximity is noted as closer or immediate. When Measured Power tends to its default value, proximity and accuracy also tend to actual distance, '1m'. When RSSI exceeds Measured power, the receiver estimates it as a position far away from the iBeacon since it is receiving weaker signals. These results are supporting the general theory, and positioning constrained explained in section 3.2, i.e., proximity estimation by receiver varies with the change in Measured Power value. This performance measure is beneficial in configuring the iBeacon in a cabinet with interference or obstructions. For example, if the receiver is showing accuracy 2.91m (as in Table 2) at an actual distance of 1m, its 'MeasuredPower' is to be set to a lower value than -58dB.

5.4 Effect of obstacles on accuracy and RSSI

The fourth set of experiments in this work was aimed at the evaluation of the effect of objects on the line of sight from iBeacon to the receiver. The transmitter was placed at the height of 1m from the ground, and the receiver is held on the same line of sight facing the transmitter. The observed values, RSSI and accuracy, were recorded from 1m to 7m at a gap of 1m. The experiments were started with an ideal environment having no interference, and repeated by placing iBeacon in a cardboard box and obstructing the direction of sigma line with a human being at less than 1 m from iBeacon. These experiments were first conducted with an iPhone SE as iBeacon1, and repeated with iPhone 6s as iBeacon2; in both cases, the receiver was set as iPad mini 4. The observed values are summarized in Table 3; for each case, values are tabulated as RSSI value with accuracy in the bracket.

Table 3. Effect of obstructing objects on RSSI and Accuracy

Distance (m)	**Without Obstructions**		**Placing iBeacon in a Card-board box**		**Human-being on line of sight**	
	RSSI(Accuracy)		***RSSI(Accuracy)***		***RSSI(Accuracy)***	
	iBeacon1	iBeacon2	iBeacon1	iBeacon2	iBeacon1	iBeacon2
1	-56(1.03)	-56(1)	-60(1.65)	-60(1.62)	-61(2.16)	-63(2.6)
2	-59(1.56)	-58(1.12)	-61(2.12)	-62(2.34)	-68(4.39)	-67(4.25)
3	-62(2.15)	-60(1.65)	-65(3.2)	-64(2.85)	-74(9.15)	-72(7.1)
4	-66(3.58)	-65(3.14)	-68(4.54)	-69(4.9)	-81(15.2)	-80(16.17)
5	-69(5.23)	-69(5.12)	-74(8.3)	-73(8.58)	-71(7.3)	-73(9.93)
6	-62(2.22)	-62(2.4)	-68(4.77)	-66(3.45)	-75(8.24)	-72(8.13)
7	-63(2.49)	-58(1.6)	-65(3.25)	-65(3.15)	-69(4.43)	-69(5.33)

RSSI values and accuracy values were observed to be changing due to room layout change, as summarized in Table 3. Accuracy results show significant variation from the actual distance after 5m in all cases. These experiments were conducted in a corridor with a thick brick wall on one side up to 5m, which expands to a more spacious location after that. Another observation is that even though iBeacon was completely enclosed in a cardboard box, no significant change observed in RSSI and accuracy values. The third case of this experiment was human intervention. As discussed in section 3.2, path loss due to signal block is evident from observed results, with a drastic decrease in RSSI and corresponding error increase in accuracy. For iBeacon1, direct RSSI at 4m was -66, whereas it is changed to -68 for case 2, and -81 for situation 3. The similar variation observed in the case of iBeacon2 also.

5.5 Discussion

It is clear from the literature and experimental results that beacon ranging varies according to the change in room layout, interference, beacon height, and obstacles. Proximity follows an inverse-square relationship [33] between distance and signal strength. The experimental study justifies all these general constraints with the measured values. iBeacon is a one-way transmit device. It cannot receive the signals from the iPhone or other devices. To provide two-way communication, the sender and receiver in this app are provided with a BLE transmission facility, even though the threat of the interference and erratic measurements exists in the environment. Restricted or optimized usage of the devices in an office environment can solve this limitation to some extent. In a typical hardware beacon, there will not be any internet connection, and beacon ranging is available only when the connected app is in foreground. In contrast to this, the proposed app as a single suite provides integrated internet services like geofencing, chat and push notifications, along with proximity services. In the case of communication disruption among team leaders and members, mutual alert mechanisms through push notifications can be used to wake up the other device.

The major limitation of the proposed app is battery drainage while keeping the app active all the time in the foreground. In regular use, around two years of battery life can be expected for an iBeacon due to its low power consumption. However, it depends on transmit power and notification interval [7][33]. Optimized and restricted usage of the app, as discussed earlier, is applicable in the context of battery saving also. Since the main focus of this app was iBeacon integration, only the bare minimum team management services were included in it. In future developments, additional features like video chat/conference, file sharing, presentations, meeting minutes, inter-team communication, etc., can be added without affecting the app complexity.

6 Conclusion

A smart device based team management application to provide a better environment for team communication and task monitoring in an organization is implemented and analyzed in this work. The entire team can detect the proximity of the members through the iBeacon services provided by the app. In addition to the proximity features, the minimum required project management services like task management, employee tracking, instant chat, employee log, and status report are also implemented. It is cost-effective and efficient in the sense that the proposed app completely avoids the installation of dedicated and static hardware iBeacon. Feasibility study and analysis of the proposed iBeacon service reveals certain constraints while using the app in an office environment. In future work, this can be expanded to multiple team connections solving the proximity limitations, and the whole group tasks can be separately maintained for the ease of usage. The proposed work can be expanded to a complete suite of communication collaboration system for project management with additional features like presentation, video chat, file sharing, etc in the future.

7 References

[1] A. J. Gilbert Silvius, Celine M. Silvius (2015), Exploring functionality of mobile applications for project management, Procedia Computer Science, 64:343 – 351. https://doi.org/10.1016/j.procs.2015.08.498

[2] Sanchez, M. A. (2018), How Internet of Things Is Transforming Project Management, In Z. Gontar (Ed.), Smart Grid Analytics for Sustainability and Urbanization, Hershey, PA: IGI Global, pp. 73-102. https://doi.org/10.4018/978-1-5225-3996-4.ch003

[3] Pulse of the Profession (2018), Success in Disruptive Times- Expanding the Value Delivery Landscape to Address the High Cost of Low Performance, PMI, [online] Available at:https://www.pmi.org/learning/thought-leadership/pulse/pulse-of-the-profession-2018#

[4] Matthew Finnegan (2019), Task management apps: Collaborative project tracking tools for the digital workplace, ComputerWorld, April 3, [online] Available at: https://www.computerworld.com/article/3386418/task-management-apps-collaborative-project-tracking-tools-for-the-digital-workplace.html

[5] Paul Chaney(2018), 15 Ways Small Businesses Can Use Beacons and Geofencing to Attract and Retain Customers, December 24, Technology Trends, [online] Available at: https://smallbiztrends.com/2017/01/location-based-marketing-beacons-geofencing.html

[6] Zhu, Lingling & Yang, Aolei & Wu, Dingbing & Liu, Li. (2014), Survey of Indoor Positioning Technologies and Systems, Communications in Computer and Information Science, 461:400-409. https://doi.org/10.1007/978-3-662-45283-7_41

[7] ChoonSung Nam, Dong-Ryeol Shin (2018), Vehicle location measurement method for radio-shadow area through iBeacon message, International Journal of Distributed Sensor Networks, 14(11). https://doi.org/10.1177/1550147718812578

[8] iBeacon and Bluetooth Beacon Market: Recent Industry Trends, Analysis and Forecast 2026, Digital Journal. [online] Available at: http://www.digitaljournal.com/pr/3940756

[9] K. E. Jeon, J. She, P. Soonsawad, P. C. Ng (2018), BLE Beacons for Internet of Things Applications: Survey, Challenges, and Opportunities, IEEE Internet of Things Journal, 5(2):811-828. https://doi.org/10.1109/jiot.2017.2788449

[10] Apple Developer website, Getting Started with iBeacon-Version 1.0, June 2 2014, [online] Available at: https://developer.apple.com/ibeacon/Getting-Started-with-iBeacon.pdf. https://doi.org/10.1017/cbo9780511583926.003

[11] Apple documentation, Turning an iOS Device into an iBeacon Device, [online] Available at:https://developer.apple.com/documentation/corelocation/turning_an_ios_device_into_an_ibeacon_device. https://doi.org/10.1007/978-1-4842-1527-2_6

[12] Nic Newman (2014), Apple iBeacon technology briefing, Journal of Direct, Data and Digital Marketing Practice 15:222–225. https://doi.org/10.1057/dddmp.2014.7

[13] Claire Swedberg(2017), London Airport Enables Proximity Apps With BLE Beacons, RFID Journal, [online] Available at:https://www.rfidjournal.com/articles/view?16214

[14] Yang J., Wang Z., ZhangX. (2015), An iBeacon-based indoor positioning systems for hospitals, International Journal of Smart Home,9:161-168. https://doi.org/10.14257/ijsh.2015.9.7.16

[15] Charles Atherton(2019), Beacons: A tool for 21st Century teaching and learning?, Research in Learning Technology, 27. https://doi.org/10.25304/rlt.v27.2127

[16] Riya Lodha, Suruchi Gupta, Harshil Jain, Harish Narula(2015), Bluetooth Smart based Attendance Management System, Procedia Computer Science, 45:524 – 527. https://doi.org/10.1016/j.procs.2015.03.094

[17] Monica Laxman Uttarwar, Arun Kumar, Peter H. J. Chong(2017), BeaLib: A Beacon Enabled Smart Library System, Wireless Sensor Network, 9:302-310. https://doi.org/10.4236/wsn.2017.98017

[18] Yang Yang, Zhouchi Li, K. Pahlavan(2016), Using iBeacon for intelligent in-room presence detection, 2016 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), San Diego, CA, pp. 187-191. https://doi.org/10.1109/cogsima.2016.7497808

[19] Santhosha Rao, Smitha A, Kunal Kulkarni, Smart Phone based Cost Effective Visitor Management System for Smart Offices, International Journal of Interactive Mobile Technologies (iJIM), 12(6):112-123. https://doi.org/10.3991/ijim.v12i6.9476

[20] M. Choi, W. Park, I. Lee (2015), Smart office energy management system using Bluetooth low energy-based beacons and a mobile app, 2015 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, pp. 501-502. https://doi.org/10.1109/icce.2015.7066499

[21] Estimote use case, All Hours - an attendance tracking software with beacons, [online] Available at: https://community.estimote.com/hc/en-us/articles/203844928-All-Hours-an-attendance-tracking-software-with-beacons

[22] staffNearby App website: https://www.staffnearby.com/#features

[23] Beacon Based Task Management Android App, August 2019. https://www.beaconzone.co.uk/blog/beacon-based-task-management-android-app/

[24] Jeongyeup Paek, JeongGil Ko, Hyungsik Shin (2016), A Measurement Study of BLE iBeacon and Geometric Adjustment Scheme for Indoor Location-Based Mobile Applications, Mobile Information Systems, 2016: Article ID 8367638, 13 pages. https://doi.org/10.1155/2016/8367638

[25] F. Zafari, A. Gkelias and K. K. Leung 2017), A Survey of Indoor Localization Systems and Technologies, IEEE Communications Surveys & Tutorials, 21(3):2568-2599. https://doi.org/10.1109/comst.2019.2911558

[26] Pavel Kriz, FilipMaly, Tomas Kozel (2016), Improving Indoor Localization Using Bluetooth Low Energy Beacons, Mobile Information Systems, 2016: Article ID 2083094, 11 pages. https://doi.org/10.1155/2016/2083094

[27] Jun-Ho Huh, Kyungryong Seo(2017), An Indoor Location-Based Control System Using Bluetooth Beacons for IoT Systems, Sensors, 17(12), 2917. https://doi.org/10.3390/s17122917

[28] Bo Xiang (2017), Application of Ranging Difference Location Algorithm in Wireless Sensor Network Location, International Journal of Online and Biomedical Engineering (iJOE) 13(7). https://doi.org/10.3991/ijoe.v13i07.7285

[29] Choosing the Transmitted Power, [online] Available at: https://www.beconzone.co.uk/choosingthetransmittedpower

[30] The complete guide to website wireframe design - Chapter 8, Beginner's guide to UI sketching, Prototyping And Wireframing, July 09, 2019, [online] Available at: https://www.justinmind.com/blog/from-a-mere-wireframe-to-a-final-website-design/

[31] Apple Developer Documentation, https://developer.apple.com/documentation/

[32] Ron Kliffer(2018), Alamofire Tutorial: Getting Started, [online] Available at: https://www.raywenderlich.com/35-alamofire-tutorial-getting-started

[33] Estimote developer docs, Part 3: Ranging beacons, [online] Available at: https://developer.estimote.com/ibeacon/tutorial/part-3-ranging-beacons/

8 Author

Sindhumol S. Ph.D.**,** is an Associate Professor in DDU Kaushal Kendra, Cochin University of Science and Technology, Kochi, INDIA. Her main research interests include image/video processing, pattern recognition, iOS app development, and IoT.

Article submitted 2019-09-15. Resubmitted 2019-11-19. Final acceptance 2019-11-19. Final version published as submitted by the authors.