Beacon - Requirements, Vision, and Scope

Motivation/Opportunity

People have a limited ability to stay connected to social events going on around them. There is no seamless way to see what events are currently going on near your current location. Additionally, there is no way for small event planners to advertise to potential attendees in their area, while their event is occurring. BEACON solves this problem by allowing users to pin their event location and information to a shared map that all users can view in real-time.

This app will separate itself from other platforms, such as the event planning system offered by Facebook, by not focusing as much on the planning aspect of social events. Instead, BEACON will try and encourage on-the-fly decision making, by focusing on nearby events that are occurring at the moment.

Problem Statement

The problem of	Spontaneously finding/planning events
affects	1) People who have no plans and are looking for fun, nearby activities/events to attend. 2) People who are hosting an event and want people to attend it.
the impact of which is	1) Missing nearby events that you are unaware of 2) Having few/no people attend your event
a successful solution would be	An application that allows user to search for nearby events/create events for others to find.

Product Position Statement

For	The public
Who	1) Want to find a nearby event 2) Want to broadcast an event they are hosting
Our System	Is an android app that communicates with a server
That	Shows events around the user's location in real-time, and allows users to create their own events.
Unlike	Other applications where you plan events ahead of time.
Our Product	Allows for spontaneity and focuses on proximity.

User Demographics

Searching Users: Users who are looking for something to do will be able to easily search for nearby events by type, name, tags, etc. The app will be designed to be immediately easy to use and will not require any complex configuration. Users who search for events will want the app to operate quickly and show a map of nearby events.

Hosting Users: Users who want to share events they are hosting will be able to create their events easily by providing basic information such as event type, location, duration, etc. The interface for creating events will be simple and user-friendly. Business owners will be able to register as businesses and broadcast their official events. Users who are hosting events will want the app to be a secure and responsive platform for sharing information about their public/private events.

Feature List

1. Real-time mapping of events and hangouts occurring within a local vicinity
    a. BEACONs overlayed on google map
    b. Shows user's current location
2. An option for the user to plan events and hangouts for future dates and make them private/ public
    a. '+' button for easy beacon creation
    b. Public events can be seen by any nearby user
    c. Private events can only be seen by those with a direct link to the event page
3. Allows for searching and sorting
    a. keyphrase
    b. name
    c. distance
    d. tags
    e. relevance
4. A system for businesses to advertise their events
5. User interface to allow the app to be tailored to the user’s interests
6. Ability for users to post comments/photos to an event listed through the app
7. Persistent Beacons (e.g. Stanley Park), which have public posts and feed.
8. Events information
    a. Title
    b. Event Description
    c. Location (for building/room etc)
    d. Time
    e. Media related to event
    f. Tags/Categories of event

Constraints

The application will be majorly constrained by the Firebase platform. The price associated with using this service will be charged based on a “as-you-go” basis. So, as BEACON grows, so will its costs.

Furthermore, if upgrades want to be made that are outside of the capabilities supported by Firebase (for whatever reason), it would be very difficult. The vital features of the app, as well as the majority of the backend functionality, are supported by Firebase, and replacing these would require much time and effort.

Scope and Limitations

The following features will not be implemented in our product

1. User “friend lists” within the Beacon application
2. The ability to Broadcast an event more than 48 hours ahead of time
3. Direct messaging between users

Assumptions and Dependencies

1. BEACON will depend heavily on Google services, especially the services offered by Firebase, to function properly. This dependency assumes that Google will not cease to provide these services.
2. BEACON will require a minimum amount of events and users. If there isn’t a critical mass of users who actively utilize BEACON, the primary goal of the app won’t be achieved.

Use Cases

Actor	Goal
Searcher	- Search for an event - Upload a comment/photo to an event
Host	- Create an event - Edit/delete their event page
Business Owner	- Advertise an event

Identification	1 - Search for an event
Primary Actor	Searcher
Stakeholders and Interests	- Hosts - Business owners
Preconditions	- Searcher has logged into the app with a Google account
Postconditions	- User has found an event to attend
Main Success Scenario	1 - Searcher opens the Beacon app 2 - A list of nearby Beacons is displayed 3 - The Searcher selects an event that suits them 4 - The Searcher views the event page, where photos and event information is displayed
Extensions/Alternative Flows	2.1 - Searcher enters a query to filter events 2.2 - An interactive map with pins tagged to Beacon locations is displayed 4.1 - Searcher attends the event 4.2 - Searcher favourites the event 4.3 - Searcher returns to list of events
Error Scenarios	2a - There are no nearby events 2a.1 - Searcher is prompted to create their own event 2.1a - Searcher's query returns no results 2.1a.1 - Searcher is asked to submit a new query
Open Issues	- Should a Google login be a precondition for viewing events?

Identification	2 - Upload a comment to an event
Primary Actor	Searcher
Stakeholders and Interests	- Hosts - Business owners - Other searchers
Preconditions	- Searcher has logged in with a Google account - Searcher has found an event they wish to upload a photo to
Postconditions	- Other searchers who view the Beacon page can view the comment
Main Success Scenario	1 - Searcher views an event page 2 - Searcher types a comment to post 3 - Searcher uploads content to event page 4 - Comment appears on event page
Extensions/Alternative Flows	2.1 - Searcher attaches a picture to the comment
Error Scenarios	3a - Comment fails to upload 3a.1 - Searcher is prompted to upload their comment again
Open Issues	- Does a searcher need to be near an event to upload a comment?

Identification	3 - Create an event
Primary Actor	Host
Stakeholders and Interests	- Searchers
Preconditions	- Host has logged in with a Google account
Postconditions	- A Beacon for the host's public event is created, and can be viewed by nearby users - A Beacon for the host's private event is created, and can be viewed only by users who have the shareable link
Main Success Scenario	1 - Host opens app and chooses to create an event 2 - Host specifies whether they want to create a public or private event 3 - Host enters information about their event -name -time -description -picture(s) -tags (for queries) 4 - Beacon is created
Extensions/Alternative Flows	4.1 - Beacon is created at host's current location 4.2 - Host specifies a location, somewhere near their location for the Beacon to be created 4.3 - If event is private, host receives a sharable link to the event page 4.3.1 - Host shares the link for their friends to view 4.4 - If event is public, Beacon becomes viewable to all nearby searchers
Error Scenarios	4.2a - Host attempts to create Beacon at a location outside of the allowable range 4.2a.1 - Host is prompted to enter a location within the valid range
Open Issues	- How far away from the Host's current location can a Beacon be created? - Should a user not be able to create an event with offensive words in the title or description? - Should a Host be able to set the visible radius of their Beacon? If not, how large should the view radius of a standard Beacon be?

Identification	4 - Edit an event page
Primary Actor	Host
Stakeholders and Interests	- Searchers
Preconditions	- Host has logged into the app through a Google account. - Host has created a Beacon for their event.
Postconditions	- Host's event information has been altered
Main Success Scenario	1 - Host opens app and goes to their event page 2 - Host confirms they want to edit their page 3 - Host edits the event information 4 - Host confirms changes and Beacon is updated
Extensions/Alternative Flows	3.1 - Host changes name, location, time, description, picture(s), and/or tags 3.2 - Host deletes comment(s) on the event page 3.3 - Host cancels the event
Error Scenarios	2a - The host is not the creator of the event in question 2a.1 - No option for editing the event will be available to them 3.1a - Host attempts to change the location of the event to somewhere outside of the allowable range 3.1a.1 - Host is prompted to enter a location within the valid range
Open Issues	- Should the host be able to change their event from public to private, or vice versa? - Should there be a limit to the number of photos the host can upload to their event page?

Identification	5 - Advertise an event
Primary Actor	Business Owner
Stakeholders and Interests	- Searchers
Preconditions	- Business owner has logged in with a Google account, and is the creator of the event page in question - Business owner has paid for advertising
Postconditions	- Business owner has broadcast their event to a wider radius - Business owner's Beacon has become distinguishable from other Beacons
Main Success Scenario	1 - Business owner opens app and goes to their event page 2 - Business owner specifies broadcast radius 3 - Business owner broadcasts their event 4 - Beacon is updated
Extensions/Alternative Flows	4.1 - Beacon pin for the event gets distinguishable by color as specified by the business owner
Error Scenarios	2a - Business owner attempts to broadcast to a radius larger than allowed 2a.1 - Business owner is prompted to specify a valid broadcast radius
Open Issues	- For advertised events, should the app send push notifications to searchers within the radius?

Non-Functional Requirements

Performance Requirements:

- The app should be usable when offline and should notify the user if certain features require a stable internet connection. If the user wants to upload event data, the app should store that data locally and push it to the server automatically when a connection is restored.
- The app should be fast and responsive even when there are a large number of nearby events. This should be achievable through smart UI design and data caching. (should we add Scalability Requirements and tie it to this?)

Safety/Censorship Requirements

- Inappropriate content should be taken down as soon as possible, the application should filter out and destroy beacons with inappropriate content according to some kind of moderator.
- A platform for the police to monitor “suspicious events”

Security Requirements

- Users that want to host a private event should be able to get a secure, shareable link that they can share with their friends without worrying about the event’s exposure to the public. This means that the people without the Beacon link cannot see the event on the app even if they are nearby.
- Users should be able to hide details that they don’t want others to see, this can be achieved by allowing event hosts to provide as many or as few contact details as they want.
- Payment for advertising will have to be handled carefully, which will necessitate using a secure well-established third party to add in-app payments.

Software Quality Attributes

- We anticipate that as we build our application, we will frequently need to revise our database schema, both for our firebase database and our local database. For this reason, the code base should not depend heavily on the structure of the databases, this way as we change our database the amount of code that needs to be refactored can be kept to a minimum.