

# LifeSpeeder

## A Web and Mobile Platform for Events Location

Pedro J.S. Cardoso<sup>1</sup>, Jânio Monteiro<sup>1,2</sup>, José dos Santos<sup>1</sup>,  
Natália Baeza<sup>1</sup>, and Sérgio Tarazona<sup>1</sup>

<sup>1</sup> Institute of Engineering (ISE), University of Algarve, 8005-139 Faro, Portugal  
{jmmonteiro,pcardoso}@ualg.pt  
<sup>2</sup> INOV, Lisbon, Portugal

**Abstract** The use of smartphones and tablets as become almost banal in these days. Smartphones, besides serving their main purpose of making and receiving calls, come to be one of the main equipments to obtain information from the Internet, using the commonly installed browsers or through the use of dedicated applications. Furthermore, several other devices are also very frequent to the majority of the modern smartphones and tablets in the market (e.g., GPS – Global Positioning System). This devices give the current systems a very high potential of usage.

One example of applicability, comes from the wish to find and navigate to events or activities which are or will soon be occurring near the user. The LifeSpeeder platform is one of the first applications in the mobile equipment market of applications which take into consideration exactly what we have just outlined, i.e., a mobile and desktop application which allows the users to locate events according with their preferences and to get help navigating to them. In this paper we briefly describe the LifeSpeeder’s front and back-end.

**Keywords:** Geographic and Temporal Location of Events, Android, NoSQL Databases.

## 1 Introduction

The growing usage of Information and Communication Technologies has recently been followed by a significant increment on the number of heterogeneous terminals that are used to access the Internet. Such terminals are currently being used in a personalized and mobile way, integrating sensors and global positioning receivers with an ubiquitous Internet access. In this field, there is a shift from location agnostic web content retrieve, to a new type of applications that take into consideration users preferences and their location to identify which data is more relevant to them.

Such shift in Internet usage gains a particular relevance in the case of events location. Information about events, like concerts or sport competitions, is by its nature associated with one or more sites and occur in one or more moments in time, after which it tends to be irrelevant. The importance of the information is

also very much dependent on the users preferences. In this field while numerous sites are specialized in some set of event types, the information they have tend to be sparsely distributed and do not automatically associated with user preferences and location. Also users tend to search and select which events to attend within hours or days before its occurrence, which requires a solution on time for their requests.

For this purpose, some applications are already available in the market. For example, the *Scoutmob* Android app [1] is a mobile guide to local deals, events, restaurants in some US cities. The *Eventbrite* Android app [2] allows to create, promote and manage events. Some cities or regions have also dedicated applications [3, 4]. The *Where To Go? GPS POI Finder* [5] provides turn-by-turn directions to the chosen destination, which include a dozen of categories. However, as far as we know none of them includes in a single application all the mentioned features, location and navigation to events with multilingual support and responding to the users preferences.

In response to the above mentioned requirements (e.g., spatially and temporally locate events according with the users preferences and navigation to them), in this paper we present the LifeSpeeder platform which combines into a web and a mobile application interfaces an event driven computational core to the localization of events. In other words, using a mobile device with an Internet connection, users are able to quickly and easily obtain lists of events that are going to happen near them or at some other temporal and spacial location. Then the mobile application calls an external application to navigate to the events. In the back-end a MongoDB database stores the data providing the support for an efficient and relatively easy implementation of the multilingual and the geolocation mechanisms [6].

The remaining document is structured as follows. The second section describes the front-end of the LifeSpeeder platform. Section 3 presents resumed details about the LifeSpeeder back-end, namely some of the used technologies. Conclusions and future work are presented in the fourth and final section.

## 2 Lifespeeder's Frontend

The LifeSpeeder project consists on the development of a platform to manage, monitor and locate events in a simple and fast way. The motivations behind the developed work starts at the fact that we are located in one of the touristic regions of Portugal. Mostly during the summer season, the Algarve's region is replete with many events and tourists that do not know well the regions, but want to get the best experience possible. Furthermore, many people decide to spend a few days in the region on the off-peak time, tacking advantage of the weather conditions and lower prices of the flights, hotels, golf courses and other infrastructures.

Although the initial idea was explored taking into consideration specially the touristic case, we came to the conclusion that most of the times the majority of the residents also don't have knowledge about the region events, since the

corresponding ads do not exist, have a small regional spread, or are scattered in different locations.

Unquestionably, the described problems are common to the large majority of the touristic regions and to day-to-day situations where dedicated platforms do not exist, where the information is scattered or where the information is relatively limited.

Taking advantage of various information technologies currently available, the LifeSpeeder platform presents a tool that can assist the users in the search for such events. The LifeSpeeder platform consists of: (a) a web application, where in addition to the search for events, the registered users can post their own events; and (b) A mobile application, implemented for the Android operating system devices, which implements the search for near (geographically and temporally speaking) events and the GPS navigation to them (using an external application).

The events/activities information, with all the necessary details, are gathered in one information system, facilitating the location according with the interest of the user, avoiding trials at multiple sites on the Web. Further details about the information system are presented in Section 3.

Furthermore, the platform offers multilingual supports and advanced search options. In other words, searches on the platform goes beyond the more classic searches since it allows the user to set its preferences, filtering and customizing the results of those queries. For example, user preferences can be based on a particular type of event (e.g., sports) or in a specific subcategory of event (e.g., football), can include parameters related to distance between the user and the event, date/time, place and language, among others.

Figure 1 sketches a global vision of the LifeSpeeder platform, where multiple devices (PC, tablets, smartphones, etc) connect to the LifeSpeeder information system, retrieving and sharing information from it. The next sections will describe in more detail the mobile and the web application.

## 2.1 Mobile Application

The growth of the mobile technologies has provided a wide variety of options and therefore its selection is of great importance. Among them, and in terms of mobile operating systems, Android OS is currently the most common operating system for smartphones, surpassing 80% market share in the third quarter of 2013 (see Table 1) [7]. It was therefore natural that the Lifespeeder's first mobile application implementation was for Android devices.

As depicted in Figure 1 the mobile application gets the information from the back-end (see Section 3). The communication steps between the mobile application and the back-end, from the first moment that the application makes a request until it receives a response, are the following:

Step 1. The mobile application uses the HTTP protocol to make a request to the LifeSpeeder Web Service (LSWS). The LSWS in turn interprets this request and verifies if the mobile device has sent its ID. If that was the



**Fig. 1.** Global vision of the LifeSpeeder platform

case, then it goes directly to step 2. If no ID was sent then the device will be registered in the system with a new generated ID, which is then sent to the device in step 4, along with the response of the request.

- Step 2. The LSWS interprets the request, and builds and executes the necessary queries to reply to the requisition made by the mobile application.
- Step 3. The database management system returns the result to the Web Service, which in turn uses these results to build a JSON document with the information.
- Step 4. With the query results formatted in JSON, the LSWS sends the document to the mobile application, thus closing the communication loop.

For example, a request with the form `URL:/webservice?UID=NULL&lat=37.027996&lng=-7.922763&events=ALL&lan=pt` corresponds to a query where a not registered or an anonymous user wants to view events of any type (`events=ALL`), that are near the location with latitude equal to 37.027996N and longitude equal to 7.922763W (`lat= 37.027996&lng=-7.922763`) and preferably in Portuguese (`lang=pt`).

As already mentioned, the answer returned to the mobile application consists of a JSON document with the events found in the vicinity of the user, plus the user ID (UID) designed to identify him. The UID, in addition of being an identifier of the device, also serves to record the actions of the user, keeping them even if he changes or uses other devices.

Regarding the localization of the users, it can be almost precisely obtained using the Global Positioning System (GPS) or approximated using IP address

**Table 1.** Worldwide smartphone sales to end users by operating system in the 3rd quarters of 2012 and 2013 (Source: Gartner, November 2013)

Operating System	3Q13 Market Share (%)	3Q12 Market Share (%)
Android	81.9	72.6
iOS	12.1	14.3
Microsoft	3.6	2.3
BlackBerry	1.8	5.2
Bada	0.3	2.6
Symbian	0.2	2.6
Others	0.2	0.4

geo-location [8]. The former solution is appropriate for a large percentage of mobile devices which already incorporate GPS, however it tends to require longer synchronization times. The latter one is quicker and works as a fallback to devices which do not have GPS. Furthermore, in Android devices, the access to the GPS is not possible using a browser, leaving in this case the single alternative of using the available APIs to get an approximated location.

Figure 2 shows six screen captures of the mobile application, namely: (a) the login form, (b) the main menu; (c) the setting menu; (d) a list of event grouped by category; (e) a list of events sorted by date; and (f) the geolocation of an event on a map. Regarding the location on the map screen capture we must refer that selecting the “Guide me” button (on the right upper corner of Figure 2 (f)) will open the default navigation application which is used to navigate to the selected event location using the Google maps and the GPS device.

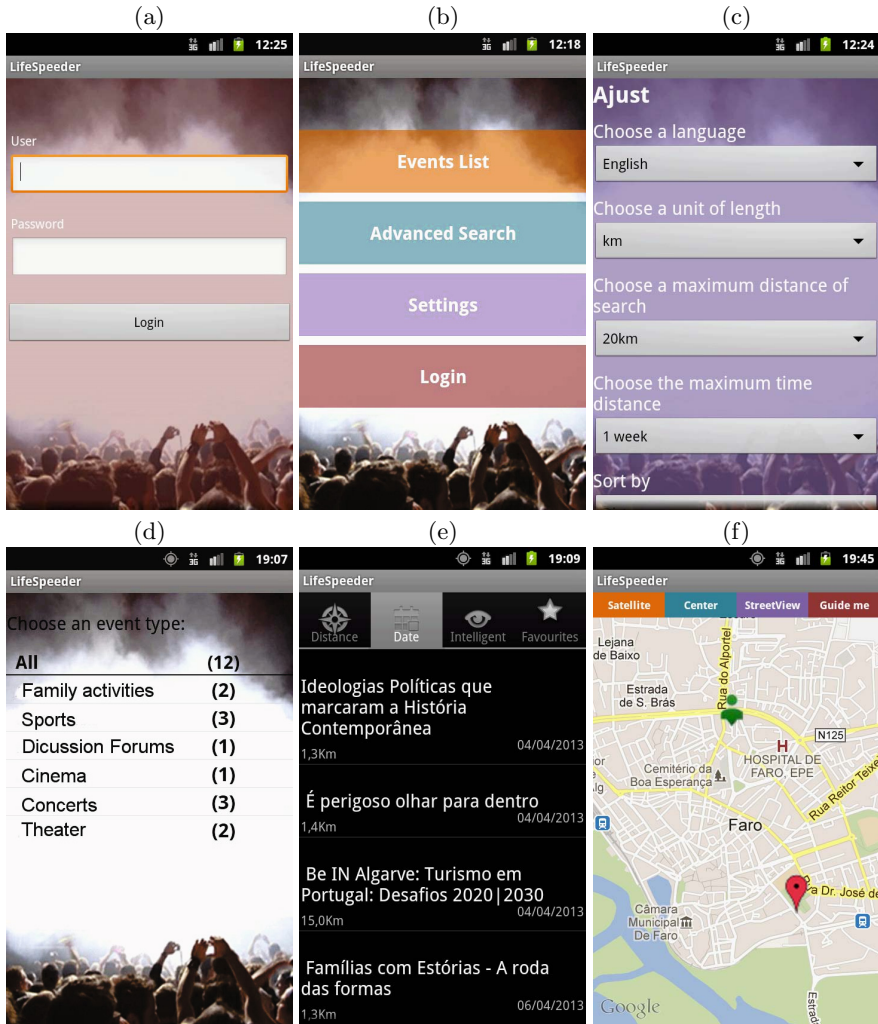
## 2.2 Web Application

The web application can be considered as the main entrance to the LifeSpeeder platform. It is from it that the events are loaded into the database, and are then provided to both applications, i.e., to the mobile applications as it has already been addressed and to the web application itself. The interface of this web application has a multilingual (currently Portuguese, Spanish and English) support, which allows to extend the horizon of usability of the platform.

As in the previous case, the web application provides the following types of searches:

**Normal search (default)** - the Geolocation API [9] approximates the location of the user based on the user’s machine IP<sup>1</sup>. The approximated location of the user and the current date is then used to search the database for spatially and temporally near events.

<sup>1</sup> For location more accurate, it is necessary that the user agrees to share its location.



**Fig. 2.** Screen captures of the LifeSpeeder mobile app: (a) Login form; (b) Main menu; (c) Setting menu; (d) List of event grouped by category; (e) List of events sorted by date; and (f) Geolocation of an event on a map

**Advanced search** - just as in the mobile application the user has the option to choose the types of events and places that he prefers. If he chooses the type and/or subtype of event but does not specifies the location, the returned list of events reflects the estimated location of the terminal. When the user chooses a location (e.g., Lisbon, Portugal), the geographical coordinates of the selected location will be obtained via the Google Maps API [10] and the events satisfying the search conditions are returned.

**Free search** - apart from the two aforementioned forms of search, you can make free query, consisting in the introduction of a string which will be looked up in the events names and descriptions.

Figure 3 shows two screen captures of the web application, namely: (a) the home page, and (b) the grouped results from a search made to the system. In the latter case, the page with the search results is divided into three sections: (i) the events grouped in different time intervals are displayed in the left side section; (ii) the events grouped in different categories are displayed in the central section, and (iii) classified and feature events on the right section.

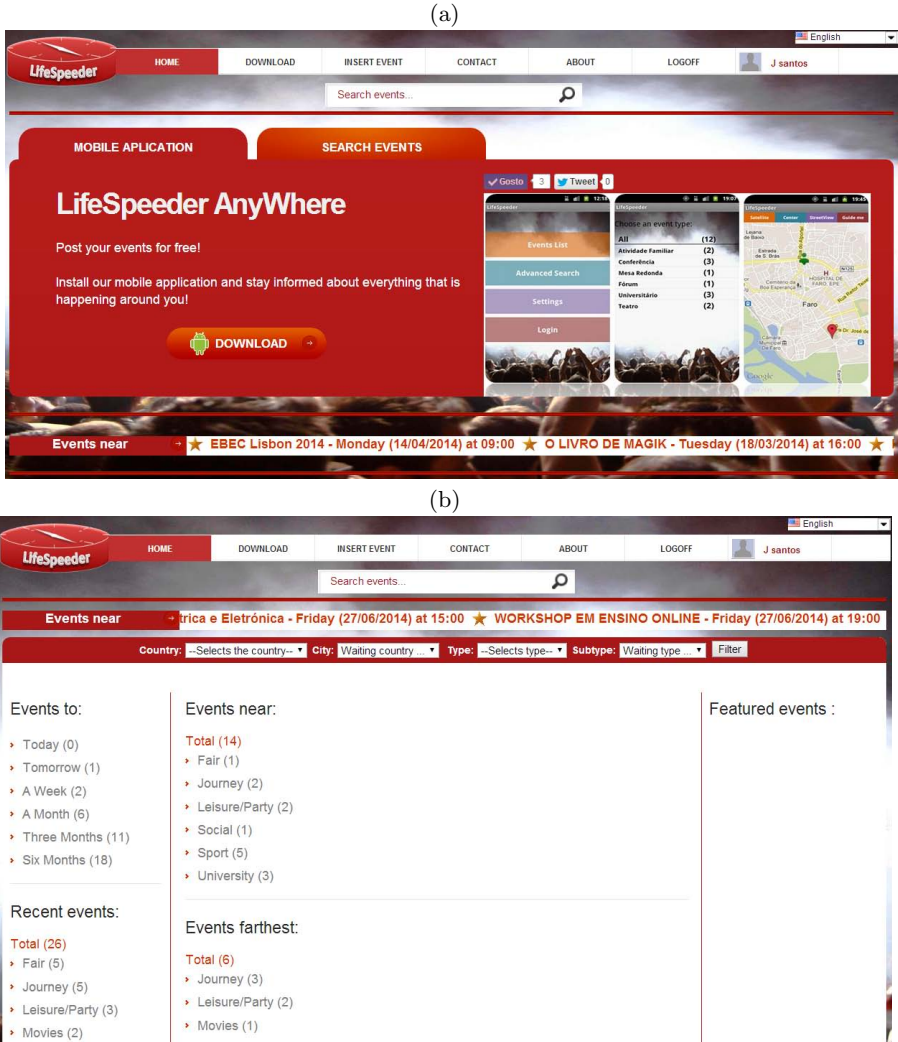
The next section will give an overview of the Lifespeeder's back-end.

### 3 Lifespeeder's Back-end

In terms of the Lifespeeder back-end, preliminary studies have shown a relatively large complexity of the data to be stored (e.g., events with distinct characteristics, events with multiple activities, events translations, multilingual support, etc). Besides, since new ideas for the application where constantly appearing, it was decided to use an iterative and incremental agile software development framework for managing the software implementation [11, 12].

An appropriate way to deal with the above mentioned conditions was to use a document oriented database solution in detriment of a relational database. In that sense, the MongoDB database was selected as the Lifespeeder datastorage [6, 13, 14]. MongoDB is an open source database, written in C++ with document oriented storage, that uses the JSON documents style, although the documents are stored in BSON. The MongoDB allows dynamic schema, since it is a schema-less database, which was a serious advantage due to the diversity of situations found in the problem. However, there are also some disadvantages such as the tendency to grow the number of fields to accommodate the variety of scenarios. This fact suggests that a modeling and analysis phase similar to the ones made for the relational schema designs is also important to avoid future drawbacks. Contrary to the modeling and analysis of the relational data bases, more centered on the data, a proper analysis for the documents structure should take into consideration the type of queries that will be done to the data and the limitation of the MongoDB's query framework. This is particularly important due to the fact that MongoDB does not support joins, and therefore it recommended to place in the same collection the data that is commonly requested together. In resume, MongoDB offers full index support, mirroring across LANs and WANs for scaling, autosharding, rich document based queries, map/reduce, and GridFS [15, 16].

Figure 4 presents a simplified example of a JSON document relative to an event. The document contains informations in three languages, the date, users comments, and the geolocation of the event.



**Fig. 3.** Screen captures of the LifeSpeeder web application: (a) Home page; (b) Grouped results from a search;



```
{
  Id_Event : 1,
  Translations : [ // array with 3 languages
    { Lang : "Portuguese",
      Title : "A UNIVERSIDADE E O MUNDO",
      Description : "'A Universidade e o Mundo' e' o tema da
        exposicao ..."
    },
    {
      Lang : "English",
      Title : "THE UNIVERSITY AND THE WORLD",
      Description : "'The University and the World' is the
        theme of the exhibition ..."
    },
    {
      Lang : "Spanish",
      Title : "LA UNIVERSIDAD Y EL MUNDO",
      Description : "'La Universidad y el mundo' es el tema
        de la exposicion ..."
    }
  ],
  Date_time : { Data : "27/09/2013", Hora : "17h30" },
  Locations:[
    {
      Location : "Biblioteca de Gambelas ",
      City : "Faro",
      Country : "Portugal",
      loc: { lat : 37.044362, lon : -7.971557}
    }
  ],
  Comments : [
    {
      User : "Marli Silva ",
      Comment : "Boa iniciativa, gosto..."
    },
    {
      User : "Beto Gomes",
      Comment : "Vou participar no evento..."
    }
  ]
}
```

**Fig. 4.** Example of a single JSON document containing information about an event

As a final word, the remaining technologies are common in this type of projects. PHP was elected as the server-side general-purpose scripting language for the web development. On the client side it was used HTML, CSS and javascript.

## 4 Conclusions and Future Work

Taking advantage of the various features and technologies currently available, it was possible to implement a tool that can assist the users to get information about events simply using a computer or some Android device.

After analyzing several storage solutions, it was concluded that a solution based on a document oriented database would be a good option. Since the events need to contain sufficient information, including geographical position, the choice of a database that natively supports these data types proved to be an asset, allowing to obtain good performances especially in queries involving research with latitude and longitude information.

On the other hand, the platform developed can be easily adapted to different situations, which opens the possibility to the creation of rebrandings. In other words, this platform can easily be adapted and customized by a company that organizes events and want to keep their customers informed about the dates and locations of the same. Another example can go through a company or agency that works with promotions, to inform customers what are the regions as well as the dates and products covered by a particular promotion.

**Acknowledgments.** This work was partly supported by the Portuguese Foundation for Science and Technology (FCT), project LARSyS PESt-OE/EEI/LA0009/2013 and project i3FR – QREN I&DT, n. 34130.

## References

- [1] Scoutmob: Scoutmob. (2014), <https://play.google.com/store/apps/details?id=com.scoutmob.ile> (retrived: October 24, 2014)
- [2] Eventbrite: Eventbrite (2014), <https://play.google.com/store/apps/details?id=com.eventbrite.attendee> (retrived: October 24, 2014)
- [3] Bordeaux Agenda: Bordeaux Agenda (2014), <https://play.google.com/store/apps/details?id=fr.bordeaux.agenda2> (retrived: October 24, 2014)
- [4] Hamburg Events: Hamburg Events (2014), <https://play.google.com/store/apps/details?id=com04131.android.hamburg> (retrived: October 24, 2014)
- [5] Where To Go?: Where To Go? GPS POI Finder (2014), <https://play.google.com/store/apps/details?id=com.ondemandworld.android.wheretogo> (retrived: October 24, 2014)
- [6] MongoDB: MongoDB (2014), <http://www.mongodb.org/> (retrived: October 24, 2014)

- [7] Butler, M.: Android: Changing the mobile landscape. *IEEE Pervasive Computing* 10(1), 4–7 (2011)
- [8] Poese, I., Uhlig, S., Kaafar, M.A., Donnet, B., Gueye, B.: Ip geolocation databases: unreliable? *ACM SIGCOMM Computer Communication Review* 41(2), 53–56 (2011)
- [9] W3C: Geolocation API specification (2012), <http://dev.w3.org/geo/api/spec-source.html> (retrived: October 24, 2014)
- [10] Google Maps API: Google Maps API (2014), <https://developers.google.com/maps/> (retrived: October 24, 2014)
- [11] Shore, J., et al.: *The art of agile development*. O’Reilly (2008)
- [12] Blankenship, J., Bussa, M., Millett, S.: *Pro Agile.NET Development with Scrum*. Apress (2011)
- [13] Han, J., Haihong, E., Le, G., Du, J.: Survey on NoSQL database. In: 2011 6th International Conference on Pervasive Computing and Applications (ICPCA), pp. 363–366. IEEE (2011)
- [14] Chodorow, K.: *MongoDB: The Definitive Guide*. 2nd edn. O’Reilly Media (2013)
- [15] Leavitt, N.: Will NoSQL databases live up to their promise? *Computer* 43(2), 12–14 (2010)
- [16] Wei-ping, Z., Ming-xin, L., Huan, C.: Using MongoDB to implement textbook management system instead of MySQL. In: 2011 IEEE 3rd International Conference on Communication Software and Networks (ICCSN), pp. 303–305. IEEE (2011)