

# Smart Space-Based Intelligent Mobile Tourist Guide: Service-Based Implementation

Alexander Smirnov<sup>\*†</sup>, Alexey Kashevnik<sup>\*</sup>, Andrew Ponomarev<sup>\*</sup>,  
Nikolay Shilov<sup>\*</sup>, Maksim Shchekotov<sup>\*</sup>, Nikolay Teslya<sup>\*</sup>

<sup>\*</sup>SPIIRAS, St.Petersburg, Russia

<sup>†</sup>University ITMO, St.Petersburg, Russia

{smir, alexey, ponomarev, nick, shekotov, teslya}@iias.spb.su

**Abstract**—The paper presents an intelligent mobile tourist guide architecture that allows tourists to get information about attractions around the current geographic location based on tourist context and estimations of other tourists. Information about attractions (images and description blocks) is extracted from different internet services (like Wikipedia, Wikivoyage, Wikitravel, Panoramio, Flickr) “on the fly” that allows to use mobile tourist guide in any region of the world and get actual at the moment information. The mobile tourist guide provides information about public transport and car sharing possibilities for the tourist with drivers nearby for comfortable reaching preferred attractions. It is based on the smart space technology that provides possibilities of personal devices and different services seamless integration. The paper also discusses services interaction in smart space that allows to implement majority of mobile tourist guide tasks in computational power devices and use personal mobile devices only for results visualization.

## I. INTRODUCTION

Modern tourists have huge possibilities to search information about interesting places through the Internet. Smartphones are mainstream with active iOS and Android devices surpassing 700 million globally by now. Global Mobile data traffic is growing rapidly to an impressive share of 13% of Internet traffic in 2012 [1]. There are a lot of services and applications that allow to simplify this search, proactively provide information about interesting attractions during the trip, provide user feedback about attending places of interests, etc. In accordance with [1] at the moment German Apple Store accounted around 780.000 apps and 36.000 travel apps (category Travel) representing a market share of 4,62% of all available apps.

Thereby, development tourist services and applications is a popular at the moment. Analysis of presented at the moment applications in the market shows that there is no application that can provide a tourist access to the information sources accessible in the Internet based on his/her location, preferences, and current situation in the area (weather, traffic jams, and etc.). Hence, it is an actual at the moment task to develop a complex application that

allows the tourist to get the following information using personal mobile device before and during the trip:

- generate list of recommended attractions based on the tourist and region contexts and attraction estimations of other tourists; tourist context characterizes the situation of the tourist, it includes his/her location, co-travelers, and preferences; region context characterizes the current situation of tourist location area, it includes his/her location, co-travelers, and preferences; region context includes such information as weather, traffic jams, closed attraction, etc.
- provide information about attractions from different sources and recommend the tourist the best for him/her attraction images and descriptions;
- propose different transportation means for reaching the attraction;
- allow the tourist to rate attractions, their images, and descriptions.

Due to the mobile device restrictions: computation complexity and power consumption of modern mobile devices it is not reasonable to perform a lot of computations in a mobile device. In this case an infrastructure is needed that allows different devices interact with each other for distribution computations during solving their tasks.

The smart spaces technology [2], [3] aims the seamless integration of different device by developing of ubiquitous computing environments, where different services can share information with each other, make different computations, and interact for joint tasks solving.

The open source Smart-M3 platform [4] has been used as a core for mobile tourist guide application. Usage of this platform makes it possible to significantly simplify further development of the system, include new information sources and services, and to make the system highly scalable. The key idea of this platform is that the formed smart space is device, domain, and vendor independent.

Smart-M3 assumes that devices and software entities can publish their embedded information for other devices and software entities through simple, shared information brokers. The Smart-M3 platform consists of two main parts: information agents and kernel [5]. The kernel consists of two elements: Semantic Information Broker (SIB) and information storage. Information agents are software entities installed on mobile devices of the smart space users and other devices that hosted smart space services. These agents interact with SIB through the Smart Space Access Protocol (SSAP). The SIB is the access point for receiving the information to be stored, or retrieving the stored information. All this information is stored in the information storage as a graph that conforms with the rules of the Resource Description Framework (RDF) [6]. In accordance with these rules all information is described by triples "Subject - Predicate - Object".

The rest of the paper is structured as follows. Section 2 presents the developed intelligent mobile tourist guide architecture. Section 3 considers mobile tourist guide service interactions in the smart space. The case study is presented in section 4. Main results and findings are summarized in the conclusion.

## II. RELATED WORK

In the previous publication, related to the considered topic [7], the authors provided the description of related work in the area of tourist guides [8], [9], [10]. There are several state-of-the-art papers that evaluate different mobile tourist guides. At the moment the following systems have been considered: GoTour [11], Tourist Attractions [12], Foursquare [13], World Explorer [14], Smart Museum [15], My Tourist Guide [16], World Around Me [17].

GoTour is an Android-based mobile application for providing tourism and geographic services in Istanbul city. Application has internal attraction database and provides possibilities of searching places of interests around using the Variable Neighborhood-based algorithm. The Tourist Attractions system determines the location and implements search for places of interests nearby in Google. The Foursquare application shows places of interests around the tourist in the interactive map and provides possibilities to estimate attended places. In addition, application allows to check-in at the attraction for showing the tourist location to other application users. World Explorer shows information from Wikipedia about attractions around the tourist. Smart Museum application provides possibilities to see information about places of interests from a prepared beforehand database. Application also provides possibilities of getting information inside museums using the Internet connection and special Smart Box installed by the Smart Museum company in different museums. My Tourist Guide provides interface to the information about most popular places in the world. The information is stored in a local database. The tourist can see tips, how to get to the place,

what is better to see, etc. World Around Me is a Windows Phone 7 application that shows the user photos around the user location. Photos are automatically downloaded from Flickr and Panoramio and showed the user.

The carried out analysis of the mentioned above systems shows that they can be divided into three main groups:

- applications that implement search for information around the tourist (e.g., World Explorer uses Wikipedia database, Tourist attractions uses Google database);
- applications that have own databases with information about attractions and provide this information to the user (e.g., SmartMuseum, Triposo [18]);
- applications that collect estimation information about attraction estimations and suggest tourists if this attraction is good or not (e.g., Foursquare, Tripadvisor [19]).

Developed mobile tourist guide application implements search of information about attraction in different Internet sources. The main distinguishing feature of proposed approach is taking into account the tourist preferences and context information for providing interesting for him/her attractions at first.

## III. INTELLIGENT MOBILE TOURIST GUIDE ARCHITECTURE

Intelligent mobile tourist guide consists of the following main components (Fig. 1):

- client application installed to the user mobile device that shares tourist context with the smart space and provides the tourist results of guide application operation;
- attraction information service that implements retrieving and caching the information about attractions;
- recommendation service that evaluates attraction/image/description scores based on ratings that have been saved to internal database earlier [20];
- region context service that acquires and provides information about current situation in the considered region (e.g., weather, traffic jams, closed attractions);
- ridesharing service that finds matching routes of the tourist and preferred attraction locations with accessible in the region drivers routes; the service provides the tourist possibilities of comfort transportation to the preferred attraction [21];
- public transport service that finds information about public transport applicable to reach the preferred attraction [7].

The client application is located in the user mobile device while other services use powerful computer systems. So, the main engineering task in the mobile tourist guide development is implementation of resource-intensive operations using other mobile tourist guide services, which is located in computational servers. Thereby, the main tasks of client application are: share information about tourist context, profile, and actions; communication with smart space; provide results to the tourist; and share tourist ratings of attended attractions, browsed descriptions and images with the smart space.

The attraction information service is responsible for four main tasks. The first one is extraction attractions titles around the tourist from external sources and shares it in the smart space. The second task is caching acquired information for quick access by tourists in the same location without additional requests to external sources at nearest future (live time of cached attraction is set by guide application administrator). The third task is providing internal identifiers for attractions, which are stored in Geo2Tag platform [22] (the platform provides possibilities for quick search of identifiers by locations). The fourth task is extraction of default images for attractions that are stored in the internal database. Default images are defined by the recommendation service as the best images for an attraction based on tourists ratings. The fifth task is extraction of attraction details (lists of images and text descriptions) from external sources and sharing them with the smart space.

And the last task is to set and refresh default images in the internal database for the attractions based on the information available in the smart space from the recommendation service.

The recommendation service is responsible for ranking attractions, images, and descriptions for providing the tourist the best attractions to see and the best images and description of chosen attraction for acquaintance. It stores the following mappings in the internal database: (internal

attraction identifier, tourist context, region context -> attraction rating), (image URL -> image rating), (description URL, description text -> description rating). The internal database also keeps similarities between users, which are calculated as background process.

The region context service acquires information about tourist location area from region specific services as soon as the client application publishes the tourist context to the smart space. The context information is published to the smart space and used by other services.

The public transport and ridesharing services provide alternatives for transportation means for the tourist to reach preferred attractions. While the public transport service provides the tourist a route that describes the sequence of transportation means to reach the preferred attraction, the ridesharing service finds a driver, which goes from the tourist location to the same direction.

#### IV. SERVICE INTERACTIONS IN SMART SPACE

The following use cases have been identified and developed for the intelligent mobile tourist guide application.

- Providing list of recommended attractions to be attended in the specified location.
- Providing an ordered lists of images and descriptions describing attraction.
- Providing the information how to reach the preferred attraction by public transport or with drivers who are going to the same direction.
- Rating attractions, descriptions, and images after acquaintance.

##### A. Providing a List of Recommended Attractions

Fig. 2 shows service interaction sequence for getting a list of recommended attractions for the tourist. Every service publishes and subscribes for triples. Subscribe triples determine notifications when a required information for the service appears in the smart space.

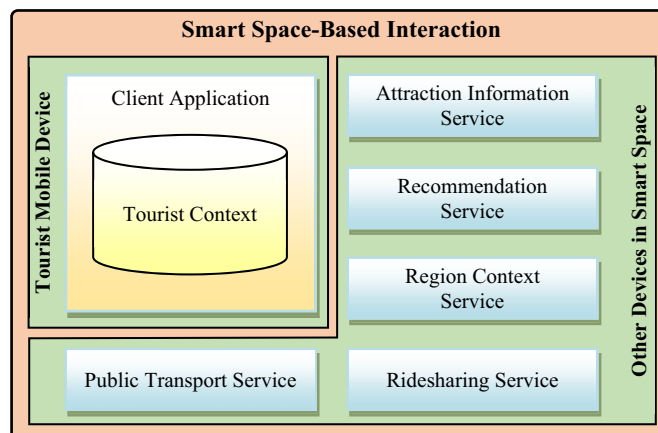


Fig. 1. General arhitecture of smart space-based intelligent mobile tourist guide

The client application shares tourist context with the smart space via publication of the following triples when the tourist searches for attractions around.

```
([user_id], "tais:is_a", "tourist")
([user_id], "geo:long2", "60.12")
([user_id], "geo:lat", "30.24")
([user_id], "tais:date_time", "2014-04-21 10:00")
([user_id], "tais:role", "traveler")
```

The client application subscribes to the following predicate:

```
([user_id], "recommended_attractions", None)
```

and gets notification every time when a list of ordered attractions around the tourist appears in the smart space. This information is shown to the tourist.

The context service subscribes to changes in a tourist context and generates the region-specific context, which is published to the smart space:

```
([user_id], "tais:weather", "sunny")
([user_id], "tais:weather_icon", "http://..")
([user_id], "tais:temperature", "20")
([user_id], "tais:traffic_jams", "average")
```

The attraction information service also subscribes to changes in a tourist context for searching the list of attractions in different sources when these changes appears in the smart space:

```
(None, "tais:is_a", None)
(None, "geo:long", None)
(None, "geo:lat", None).
```

Each attraction has unique internal identifier kept in the Geo2Tag platform [22] and can be quickly extracted by attraction coordinates. Using this identifier the attraction information service searches in the internal database default image. If it has been assigned to this attraction previously the attraction information service use the image as default for the attraction. In other case the attraction information service searches for image in Internet sources and save it as default in the internal database. The list of attractions with default images is published to the smart space:

```
([user_id], "tais:near_by", NearbyXML),
```

where NearbyXML has the following structure:

```
<?xml version="1.0" encoding="UTF-8"?>
<Attractions xmlns="http://cais.iias.spb.su/XML/tais">
  <Attraction>
```

```
    <ID>[longitude]+[latitude]</ID>
    <Name>The Hermitage</Name>
    <Distance>0,2</Distance>
    <Longitude>60,125</Longitude>
    <Latitude>30,18</Latitude>
    <DefaultImage>http://en.wikipe...</DefaultImage>
  </Attraction>
</Attraction>
...
</Attraction>
</Attractions>
```

The recommendation service subscribes to the list of attractions via the following triple:

```
(None, "tais:near_by", None)
```

and gets notification every time when the attraction information services publishes new list of attractions. After that the recommendation service queries an information about the tourist and region contexts via the queries using the following triples:

```
([user_id], "tais:date_time", None)
([user_id], "tais:role", "traveler")
([user_id], "tais:weather", "sunny")
([user_id], "tais:temperature", "20").
```

The recommendation service evaluates the attraction scores and publishes the ordered list of attractions based on acquired context information of tourist and region [20]:

```
([user_id], "recommended_attractions",
RecommendedAttractionsXML),
```

where the RecommendedAttractionsXML has the following additional tags for the "Attraction" section of the NearbyXML:

```
<Score>80</Score>
<Rating>5</Rating>,
```

where rating is the estimation of the current attraction set by the tourist last time he/she visited this attraction (can be empty) and the score is a value calculated value by recommendation service for attractions ordering.

The client application receives notification for its subscription and presents the results to the tourists using graphical interface (see sec. V).

<sup>1</sup> Hereinafter tais means <http://cais.iias.spb.su/XML/tais>

<sup>2</sup> geo:lat and geo:long predicates are defined by "Basic Geo (WGS84 lat/long) Vocabulary" and must be prefixed with "[http://www.w3.org/2003/01/geo/wgs84\\_pos#](http://www.w3.org/2003/01/geo/wgs84_pos#)"



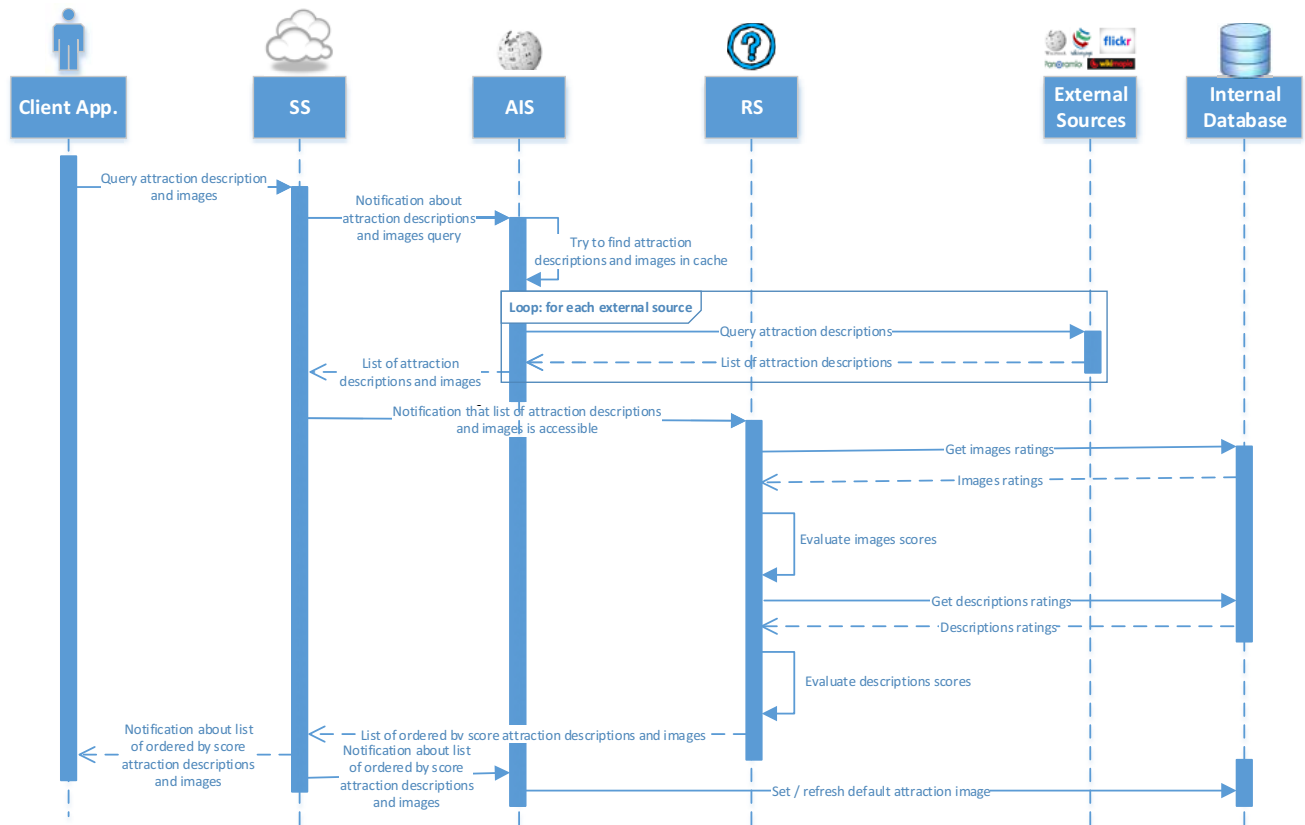


Fig. 3. UML sequence diagram of services interaction for getting attraction descriptions and images

The recommendation service subscribes to the list of attraction descriptions and images via the following triple:

*(None, "tais:attraction\_details\_bundle", None)*

The recommendation service gets images and descriptions ratings from its internal database and calculates scores [20]. Then, it publishes the list of ordered by score attraction descriptions and images:

*([user\_id], "tais:attraction\_details", AttractionDetailsXML),*

where AttractionDetailsXML has the following additional tags for the attraction section of the AttractionDetailsBundleXML:

`<Score>90</Score>`

`<Rating>1</Rating>`.

The client application receives notification for its subscription and presents the results to the tourists via graphical interface (see sec. V).

The attraction information service also subscribes to the same triple as the client application does but for all tourists. Getting this information allows storing default image for the every attraction in the internal database processed by the recommendation service. This image will be shown in

the future when the attraction appears in the list of attractions around the tourist:

*(None, "recommended\_attractions", None).*

### C. Providing Information How to Reach a Preferred Attraction

When a tourist would like to attend an attraction the client application shares the following triples:

*([user\_id], "transport:detour", 300)*

*([user\_id], "transport:delay", 5)*

*([user\_id], "transport:point", [used\_id] "dest")*

*([user\_id] "dest", "geo:long", "60.52").*

*([user\_id] "dest", "geo:lat", "30.32").*

Public transport and ridesharing services subscribe to the following triple:

*(None, "transport:point", None)*

and get notifications every time when the tourist adds a new destination point. After the route calculation the public transport service shares the following triple:

*([user\_id], "tais:recommended\_public\_transport", PublicTransportXML),*



where PublicTransportXML contains information how to reach the attraction by public transport:

```
<?xml version="1.0" encoding="UTF-8"?>
<Transportation xmlns="http://cais.iias.spb.su/XML/tais">
  <Drive arr_time="10:10" dep_time="10:00" distance="3.04"
    type="Bus 562">
    <Departure geo:lat="60.97" geo:lon="32.93"/>
    <Arrival lat="60.97" lon="32.96"/>
  </Drive>
  <Drive arr_time="10:20" dep_time="10:10" distance="4.3"
    type="Bus 56">
    <Departure lat="60.97" lon="32.96"/>
    <Arrival lat="60.69" lon="33.61"/>
  </Drive>
</Transportation>
```

The ridesharing service publishes the following triple, when it completes the calculations:

```
<[user_id], "tais:recommended_ridesharing", [{lat: 59.938,
'long': 30.360, 'alone': 'yes'}, {lat: 59.938, 'long': 30.360, 'alone':
'no', 'mp': 'yes'}]>
```

where object is the ridesharing path that defines a list of points for the tourist and driver movements. *Long* and *lat* are longitude and latitude of every point, *alone* determines if the tourist goes from this point alone or with the driver, *mp* means that this point is a meeting point.

#### D. Rating Attractions, Descriptions, and Images

After attending an attraction the tourist has a chance to estimate it. In this case the client application publishes the following triple:

```
<[attraction_id], "tais:rate_attraction", RateAttractionXML>
```

where RateAttractionXML contains rating for attraction, user, and region contexts.

```
<?xml version="1.0" encoding="UTF-8"?>
<Evaluation xmlns="http://cais.iias.spb.su/XML/tais">
  <UserID>[user_id]</UserID>
  <Context>
    <DateTime>2013-10-09 10:59</DateTime>
    <Weather>Sunny</Weather>
    <Company>Alone</Company>
  </Context>
  <Rating>4</Rating>
</Evaluation>
```

The tourist also has possibility to estimate images related to an attraction. In this case the client application publishes the following triple:

```
<[image_url], "tais:rate_image", RateImageXML>
```

where RateImageXML contains rating of the image:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<Evaluation xmlns="http://cais.iias.spb.su/XML/tais">
  <UserID>[user_id]</UserID>
  <Rating>4</Rating>
</Evaluation>
```

For estimating an attraction description the following triple is used:

```
<[description_url], "tais:rate_description", RateDescriptionXML>
```

where RateDescriptionXML has the same structure as RateImageXML.

## V. PROTOTYPE IMPLEMENTATION

Implementation of the intelligent mobile tourist guide is based on the Android OS that is one of the most popular operating systems for mobile devices today [23]. Guide is available for free downloading in Google Play Market<sup>3</sup>

Fig. 4 shows the main application screen and settings page. In the main screen the tourist can see images extracted from accessible internet sources around, clickable map with his/her location, context situation, and the best attractions around ranked by the recommendation service. By pressing “menu” button guide application allows to search information for worldwide attractions by choosing another area (country, region, and city) and access the settings page of the mobile tourist guide application. In the status bar the tourist can search for attractions worldwide.

In the settings screen (Fig. 4) the following parameters can be chosen: language, recommendations using, and smart space connection settings (for using recommendation and other services). Parameter language determines the language of the extracted information, while the interface language is determined by the Android OS settings.

Tourist can browse information about the best attractions around presented by the mobile tourist guide in the main screen and press button “More” to see more attractions (see Fig. 5).

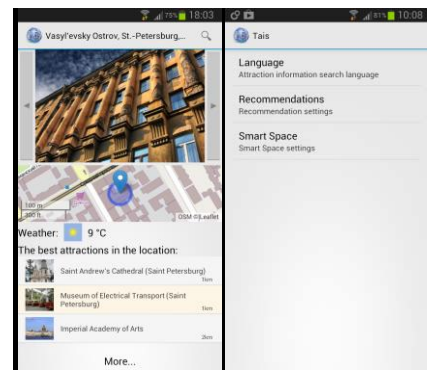


Fig. 4. Main application screen and settings page

<sup>3</sup> Mobile Tourist Guide Client for Android OS is available at <https://play.google.com/store/apps/details?id=ru.nw.spiras.tais>

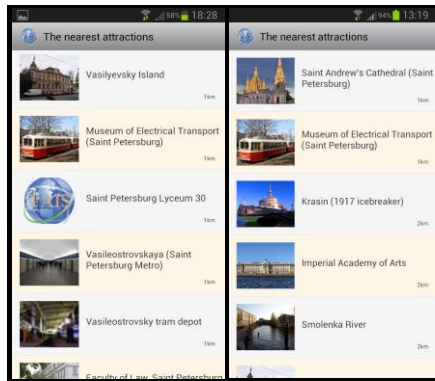


Fig. 5. List of attractions with and without recommendations

In the left screenshot a list of attractions without using recommendation service is shown (attractions are ordered by distance from the user). In the right screenshot the attractions are ordered by the recommendation service according to attraction estimations presented in Table II. Table I accumulates information about previous estimations of the tourist (John) and other tourists (Lisa, Alice, Bob) who have similar interests.

As can be seen from Table I mobile tourist guide application users (Lisa, Alice, Bob, Patrick, Jack) have

TABLE I. EXAMPLE OF ATTRACTION ESTIMATIONS OF MOBILE TOURIST GUIDE APPLICATION USERS

	Louvre	Eiffel Tower	Palace of Versailles	Notre Dame de Paris	Ile de la Cite	Museum of Electrical Transport	Imperial Academy of Arts	Krasin	Smolenska River	Saint Andrew's Cathedral
<b>John</b>	5	5	3	3	2					
<b>Lisa</b>		5	3	2	2	5	5		5	5
<b>Alice</b>	5	5	3	2	2	4	5	5	4	5
<b>Bob</b>	5	5	3	2	2	5	4	5		5
<b>Patrick</b>	2	1		3	4	5	2	4		2
<b>Jack</b>		4	5		5		3	4		3

TABLE II. GROUP OF GUIDE APPLICATION USERS WITH SIMILAR ESTIMATIONS AS THE TOURIST

	Museum of Electrical Transport	Imperial Academy of Arts	Krasin	Smolenska River	Saint Andrew's Cathedral
<b>Lisa</b>	5	5		5	5
<b>Alice</b>	4	5	5	4	5
<b>Bob</b>	5	4	5		5

The ridesharing service finds a driver for the tourist to reach an attraction. The tourist interactively gets an information about possible driver with his/her profiles, meeting points, meeting time, full recommendations about the route (Fig. 7, left screenshot). The path in the figure means that the tourist has to walk to the meeting point with the driver, then he/she goes with the driver, leaves the car at the drop-off point, and then walks to the attraction.

Right screenshot in Fig. 7 shows pedestrian path from the tourist location to the interesting attraction.

estimations of attractions around the tourist (Museum of Electrical Transport, Imperial Academy of Arts, Krasin, Smolenska River, Saint Andrew's Cathedral). The tourist (John) hasn't estimations for these attractions because he has never been here before. John, Lisa, Alice, and Bob have been joined to the group using information about their previous estimations of the same attractions.

Then recommendation service determine estimations of all users in this group for attractions nearby the tourist (see Table II). Based on these estimations, similarities between tourists, distances to attractions, and context information, the recommendation service sorts attractions around the tourist (see Fig. 5, the right screenshot).

The tourist has possibilities to see recommended attractions in the interactive map (Fig. 6, left screenshot). OpenStreetMap is used as a base platform for maps operations [25]. In Fig. 6, the right screenshot, images and description for attraction are shown. The tourist can leaf over images to browse them. Images are extracted from different Internet sources ranked by the recommendation service according to their estimations.

The developed context service searches for weather conditions in a user location at the moment. It uses World Weather Online [24] resource to get this information. The resource API allows to implement not more than 500

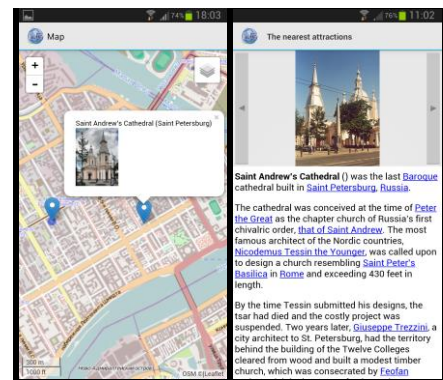


Fig. 6. An attraction in the interactive map and attraction description page



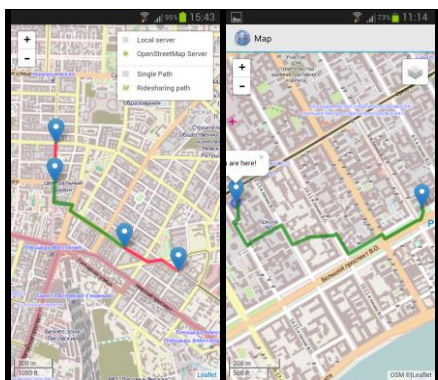


Fig. 7. Finding drivers for tourists that go to the same direction

requests per hour for free and has premium API for an increased number of requests. Weather information is updated every 3-4 hours.

## VI. CONCLUSION

The presented in the paper approach allows to develop mobile tourist guide as a distributed application that consists of several services interacting with each other through smart space technology. For the semantic interoperability, the RDF ontologies are used. Every service shares triples with the smart space according to its own ontology. The architecture allows to take in to account the tourist context and the current situation in the considered area for personalizing the list of attractions.

Implementation of the service shows that approach as applicable to the considered domain. Mobile tourist guide response time not more that few seconds for every operation. E.g., for the center of St.Petersburg (Vasilyevsky Island area) recommendation of attractions use case takes about 3 seconds for the acquiring (2,9 sec), ranking (0,1 sec) and providing to the tourist up to 50 nearest attractions.

## ACKNOWLEDGMENT

This research is a part of grant KA322 «Development of cross-border e-tourism framework for the programme region (Smart e-Tourism)» of Karelia ENPI programme, which is co-funded by the European Union, the Russian Federation and the Republic of Finland. The presented results are part of the research carried out within the project funded by grants # 13-07-00336 and 13-07-12095 of the Russian Foundation for Basic Research. This work was partially financially supported by Government of Russian Federation, Grant 074-U01

## REFERENCES

- [1] S. Wagner, T. Franke-Opitz, C. Schwartze, F. Bach, "Moblie Travel App Guide: Edition 2013 powered by ITB", *Pixell Online Marketing GMBH*, 2013, Web: [http://www.itb-berlin.de/media/itb/itb\\_media/itb\\_pdf/publikationen/MTAG\\_2013.pdf](http://www.itb-berlin.de/media/itb/itb_media/itb_pdf/publikationen/MTAG_2013.pdf).
- [2] D. J. Cook and S. K. Das, "How smart are our environments? an updated look at the state of the art", *Pervasive and Mobile Computing*, vol. 3, no. 2, 2007, pp. 53-73.
- [3] S. Balandin and H. Waris, "Key properties in the development of smart spaces", in *Proc. 5th Int'l Conf. Universal Access in Human-Computer Interaction*, Springer-Verlag, 2009, pp. 3-12.
- [4] Smart-M3 at Sourceforge, Web: <http://sourceforge.net/projects/smart-m3>.
- [5] J. Honkola, H. Laine, R. Brown, O. Tyrkko, "Smart-M3 Information Sharing Platform", in *Proc. ISCC 2010*, IEEE Comp. Soc.; Jun. 2010, pp. 1041-1046.
- [6] Resource Description Framework (RDF). W3C standard, Web: <http://www.w3.org/RDF/>.
- [7] A. Smirnov, A. Kashevnik, S. Balandin, S. Laizane, "Intelligent Mobile Tourist Guide: Context-Based Approach and Implementation", *Internet of Things, Smart Spaces, and Next Generation Networking*, Lecture Notes in Computer Science, vol. 8121, Aug. 2013, pp 94-106.
- [8] M. Kenteris, D. Gavalas, D. Economou, "Evaluation of Mobile Tourist Guides", *The Open Knowledge Society, a Computer Science and Information Systems Manifesto, Communications in Computer and Information Science*, vol. 19, 2008, pp 603-610.
- [9] J. Baus, K. Cheverst, C. Kray, "A Survey of Map-based Mobile Guides", *Map-based Mobile Services*, 2005, pp 193-209.
- [10] W. Schwinger, Ch. Grun, B. Proll, W. Retschitzegger, A. Schauerhuber, "Context-Awareness in Mobile Tourist Guides", *Handbook of Research on Mobile Multimedia, Second Edition*, 2009, p.19.
- [11] K. Al-Rayes, A. Sevkli, H. Al-Moiqel, H. Al-Ajlan, K. Al-Salem, N. Al- Fantoukh, "A Mobile Tourist Guide for Trip Planning", *IEEE Multidisciplinary Engineering Education Magazine*, vol. 6, no. 4, Dec 2011, pp. 1-6.
- [12] Google Play Market, Tourist Attractions application, Web: <https://play.google.com/store/apps/details?id=com.uknowapps.android.attractions&hl=ru>.
- [13] Foursquare application, Web: <http://foursquare.com>.
- [14] Apple iTunes, World Explorer, Web: <https://itunes.apple.com/ru/app/world-explorer-russkij-putevoditel/id381581095?mt=8>.
- [15] Smart Museum, Web: <http://smartmuseum.ru/>.
- [16] Google Play Market, My Tourist Guide application, Web: <https://play.google.com/store/apps/details?id=com.mytourguideanmyt ou&hl=ru>.
- [17] A. Vdovenko, A. Lukovnikova, S. Marchenkov, N. Sidorcheva, S. Polyakov, D. Korzun, "World Around Me Client for Windows Phone Devices", in *Proc. 11th FRUCT Conf.*, 2012, pp. 206-208.
- [18] New travel guide Triposo brings algorithms to apps, *The Independent, September*, 2011, Web: <http://www.independent.co.uk/travel/news-and-advice/new-travel-guide-triposo-brings-algorithms-to-apps-2353352.html>.
- [19] Tripadvisor, Web: <http://www.tripadvisor.ru>.
- [20] A. Smirnov, A. Kashevnik, A. Ponomarev, N. Shilov, M. Shchekotov, N. Teslya, "Recommendation System for Tourist Attraction Information Service", in *Proc. FRUCT Conf.*, Nov. 2013, pp. 140-147.
- [21] A. Smirnov, N. Shilov, A. Kashevnik, N. Teslya, S. Laizane, "Smart Space-based Ridesharing Service in e-Tourism Application for Karelia Region Accessibility. Ontology-based Approach and Implementation", in *proc. 8th Int. Joint Conference on Software Technologies*, July 29-31, 2013, Reykjavik, Iceland, pp. 591-598.
- [22] K. Krinkin, K. Yudenok, "Geo-coding in Smart Environment: Integration Principles of Smart-M3 and Geo2Tag", *Internet of Things, Smart Spaces, and Next Generation Networking*, Lecture Notes in Computer Science, vol. 8121, Aug 2013, pp. 107-116.
- [23] International Data Corporation (IDC), Press Release, Android and iOS Combine for 91.1% of the Worldwide Smartphone OS Market in 4Q12 and 87.6% for the Year, Web: <http://www.idc.com/getdoc.jsp?containerId=prUS23946013>.
- [24] World Weather Online, Web: <http://worldweatheronline.com/>.
- [25] A. Smirnov, N. Shilov, A. Kashevnik, N. Teslya, "OpenStreetMap-Based Dynamic Ridesharing Service", in *Proc. Information Fusion and Geographic Information Systems, Sixth International Workshop*, May 2013, pp. 103-118.