Information Technology & Tourism (2019) 21:413–455
https://doi.org/10.1007/s40558-019-00150-5

ORIGINAL RESEARCH

# A mobile personalized tourist guide and its user evaluation

**Ernesto Tarantino[1] · Ivanoe De Falco[1] · Umberto Scafuri[1]**

Received: 7 January 2019 / Revised: 26 June 2019 / Accepted: 1 July 2019 / Published online: 5 July 2019


**Abstract**
The paper presents an interactive electronic guide application prototype able to recommend personalized multiple-day tourist itineraries to mobile web users. The proposed application relies on an evolutionary optimizer that allows the determination, in an acceptable time, of a near-optimal user-adapted tour for each day of the visit by considering different conflicting objectives. The tour optimizer automatically plans the itinerary by selecting the sights of potential interest based on user preferences, the available visit time considered on a daily basis, opening days and hours, visiting times, accessibility of the places of interest and weather forecasting. The interactive functionalities and facilities provided by the application are illustrated along with the model used to adapt the tourist itinerary to user preferences and constraints. An experimental qualitative and quantitative evaluation has been performed to assess the validity of the guide prototype. Particular attention has been devoted to the usability of the application and its graphic unit interface along with user satisfaction.

**Keywords** Interactive mobile applications · Personalized tourist routes · Heuristics · User evaluation

## 1 Introduction

During last few years there has been a considerable evolution of tourist behavior with a constantly increasing abandonment of pre-organized tourist packages, coupled with generalized tourist routes, in favor of personalized itineraries (Hyde and Lawson 2003; Rodríguez et al. 2012). However, the large quantity of available

✉ Ernesto Tarantino
ernesto.tarantino@icar.cnr.it

Ivanoe De Falco
ivanoe.defalco@icar.cnr.it

Umberto Scafuri
umberto.scafuri@icar.cnr.it

[1] Institute of High Performance Computing and Networking, National Research Council of Italy, Naples, Italy

information and the wide range of alternatives to be considered when organizing a trip mean that the planning of a personalized tourist visit requires an enormous amount of time and effort. In fact, the tourists themselves have to gather information about the different Points of Interest (*POI*s) and then plan the itinerary to ensure that they are able to visit all the places they have selected and so satisfy all their specialized interests. This task requires complicated decisions involving different kinds of constraints (Brown and Chalmers 2003) and multiple conflicting objectives and alternatives (Kou et al. 2012, 2014; Kou and Lin 2014). Given the complexity of the problem, even an extensive analysis does not assure the selection of the best alternative since it is very difficult to simultaneously satisfy conflicting criteria and user constraints. For these reasons, tourists have increasingly tended to adopt intelligent guide systems designed to exploit new technologies to support them in the planning of their trips. Particularly important in this area are the multimedia mobile technologies which offer a constantly expanding supply of dynamic and interactive information in order to design customized experiences. In the last few years, several Personalized Electronic Tourist Guides (PETGs) based on mobile technologies have been developed to better assist tourists in satisfying their particular requirements and in ensuring a more efficient use of time.

In this paper we illustrate the design of an interactive PETG application able to arrange personalized multiple-day walking tours with contrasting objectives and different constraints related to both the tourist and the sites to visit, and adapted to user preferences. This problem can be considered as an extension of the standard single-objective Team Orienteering Problem with Time Windows (TOPTW) (Vansteenwegen et al. 2009b; Gunawan et al. 2016) because it deals with multiple objectives and a great number of constraints. Accordingly, it can be modeled as a Multi-Objective TOPTW (MO-TOPTW). In reality, the implementation of this application is currently in its prototypal stage. Since TOPTW is a complex combinatorial optimization problem belonging to the class of NP-complete problems (Souffriau et al. 2008; Righini and Salani 2009; Vansteenwegen et al. 2011a), MO-TOPTW is a task at least as complex as TOPTW. Recent research demonstrates that heuristic and metaheuristic approaches appear to be the only feasible way to face TOPTW problems efficiently (Vansteenwegen et al. 2009a; Montemanni et al. 2009; Tricoire et al. 2010; Montemanni et al. 2011; Gambardella et al. 2012; Labadie et al. 2012; Lin and Yu 2012; Hu and Lim 2014; Verbeeck et al. 2014; Gavalas et al. 2015b). The core of our PETG is represented by an evolutionary optimizer, relying on a multiobjective evolutionary algorithm (Deb 2001), which is a viable means of attaining a suboptimal solution to such a complex problem in an acceptable computational time. The tour optimizer automatically generates a personalized tour based on data relating to the user and the visit and the *POI* models considered (Gavalas et al. 2012). A description of the features included in all these models will be reported in Sect. 3.

In our previous papers (De Falco et al. 2015, 2016) we aimed at illustrating the architecture of our multiobjective evolutionary optimizer, as well as providing details of its implementation. The main contributions of this paper instead are related to the mobile application prototype and can be summarized as follows: (i) an illustration of the interactive functionalities and the facilities offered by the application, such as the multiple visualization modalities and the augmented reality; (ii) the description of its ability to plan an itinerary adapted to user preferences and dynamic constraints; and (iii) a report on an experimental qualitative and quantitative evaluation of the user experience with particular attention to the usability of the prototype and its graphic unit interfaces (GUIs) together with user satisfaction.

The quality of the application is ascertained by benchmarking the user experience and identifying the principal features that contribute to its positive evaluation.

Differently from other PETGs discussed in the literature, it is worth noting that our guide is innovative because it is able to deal with a greater number of contrasting objectives and possesses certain features that are absent from other referenced PETGs as detailed in Sect. 2.1.

## 2 Related research on personalized tourist guides

Over the last few years, personalization research has been a prominent domain in cultural heritage information. However, such a personalization necessitates a system that is capable of modeling users according to their personal preferences and contextual aspects. A survey on this research area can be found in Ardissono et al. (2012).

The customization challenges tackled by researchers have given rise to the design of flexible, efficient, and user-friendly PETGs for mobile devices. In Kenteris et al. (2011) a review of several more or less sophisticated guide applications is outlined while a survey on mobile recommender systems in tourism can be found in Gavalas et al. (2014a). A detailed overview of smart e-tourism recommender systems that employ artificial intelligence techniques is reported in Borràs et al. (2014). The paper takes into account the different kinds of GUIs and the functionalities supplied by these systems along with the diversity of recommendation algorithms.

An overview of the most significant algorithmic implementations of existing tourist guides, with a focus on those most recently developed, is reported below.

In Diosteanu et al. (2011), the authors propose a mobile application relying on a hybrid multiobjective genetic algorithm for a TOPTW problem. Such an algorithm integrates an enhanced heuristic proposed in Vansteenwegen et al. (2011a) to plan a path from a set of sites each characterized by a score that measures its attractiveness. Additionally, the start and arrival time, the opening and closing time, and the visit duration are considered.

Vansteenwegen et al. (2009b) and Souffriau et al. (2009) present an advanced mobile tourist guide relying on an interest profile, current attraction values and trip information, capable of suggesting a route among a near-optimal and practicable selection of sights. The heuristic solves a TOPTW problem by exploiting a combination of guided local search metaheuristics.

Garcia et al. (2009) propose an intelligent routing system for a PETG capable of addressing a TOPTW entailing multiple routes to transfer from different locations. This system recommends a personalized tour merging information about the tourist profile and constraints with current knowledge of local attractions, public means of transport and weather forecasting. The procedure involves an iterated local search metaheuristic method (Lourenco et al. 2002).

In Kenteris et al. (2010) the authors also propose a heuristic procedure for the generalization of a TOPTW problem to plan personalized daily itineraries for mobile tourist guides. However the approach presents some assumptions, such as the fact that the *POI*s are always open during the available visit time, that each daily tour starts and finishes at the same place, that only the shortest length path is considered, and the tourist is presumed to move at a constant speed.

Vansteenwegen et al. (2011b) suggest a PETG to address an extension of the TOPTW problem by introducing multiple TWs per *POI* where TWs can be diverse on different days (Multiple TOP/TW). This PETG relies on a metaheuristic algorithm, namely a greedy randomized adaptive search procedure (Feo and Resende 1989). The guide is able to propose a multiple-day personalized tour on the fly by considering a set of *POIs* with a score, the visit duration and the opening hours of the attraction for each day combined with route constraints and tourist interests. Furthermore, it is possible to schedule only one lunch break with no fixed location or exact timing.

Garcia et al. (2011) describe the e-Tourism system that is a tourist web-based recommender system that supports a user or group of users in the planning of a personalized tour by handling different information sources, such as opening hours of attractions, distances between the places to visit and visiting times.

In Garcia et al. (2013), the authors address a smarter approach to a PETG able to tackle in real time a Time-Dependent TOPTW (TDTOPTW), which also considers the time needed to move between locations. Specifically, they consider the options of public transport transfers in addition to walking in the planning of the tour relying on tourist preferences. The solution strategy, conceived to address the TDTOPTW, relies on the algorithm proposed in Vansteenwegen et al. (2009b) for the TOPTW and exploits an iterated local search technique (Lourenco et al. 2002).

A web/mobile recommender-based tourist guide, GuideMe, that allows for user consultation, and the publication and recommendation of tourist locations, is described in Umanets et al. (2014). The novelties of this guide in comparison to other recommender-based guides regard the specific set of offered options and the interaction with social networks.

A mobile tourist planning application, namely PSiS Mobile, able to assist a tourist during her/his holiday, is outlined in Anacleto et al. (2014). It provides recommendations for *POI*s to visit relying on tourist preferences and user and site context. This tool also works as a trip diary by registering the transfers and activities to remind the tourist about his/her holiday and his/her degree of satisfaction with the experience.

In Gavalas et al. (2015a) the authors advance a web/mobile tourist tour planner which also considers the option of using means of public transportation. The planner solves a TDTOPTW problem by employing an efficient heuristic that plans personalized tours by deriving a near-optimal sequencing of *POI*s along recommended routes minimizing the waiting time at transit stops, and taking into account several restrictions and *POI* properties, with the aim of exploiting the available time for sightseeing as efficiently as possible.

Brilhante et al. (2015) introduce TripBuilder, an unsupervised skeleton for the planning of personalized itineraries. The task of devising a personalized tour is modeled as an instance of the Generalized Maximum Coverage problem (GMC), with the aim of ensuring the optimal visit, taking into account the tourist's personal interests, holiday duration and budget.

Recently, in Muccini et al. (2017), the authors suggest a mobile guide directly running on a device that implements a recommender system algorithm to address a multi-site congestion management issue with the aim of maximizing the tourist experience. Such a guide is based on an enumeration-based heuristic algorithm that solves a TDOP problem (Verbeeck et al. 2014).

Moreover, a context-aware mobile city guide application, relying on an iterated local search meta-heuristic, that solves a Mixed TOPTW (MTOPTW) is introduced (Gavalas et al. 2017). Such a variant models a Tourist Trip Designer Problem (TTDP) (Gavalas et al. 2014b) where profit is associated not only with the *POI*s but also with the itineraries. The associated guide enhances the version developed in Gavalas et al. (2015a) by including in the tour the scheduling of scenic walking routes.

Finally, Kotiloglu et al. (2017) present a Multi-Period Multi-Constraint Orienteering Problem with Multiple Time Windows solved via an iterated Tabu Search algorithm. Their technique combines mandatory points relying on tourist preferences with a further set of optional items recommended by the guide in order to maximize user satisfaction during a multi-day tour. Other variants of itinerary planning according to personal interests and preferences in various conditions can be found in Sylejmani et al. (2017), Expósito et al. (2019), Fogli and Sansonetti (2019).

### 2.1 Comparison of PETGs providing walking itineraries

Most of the tour planning algorithms of the PETGs previously illustrated only consider one objective, namely the maximization of the total score of the visited locations. The MO-TOPTW algorithm tackles five contrasting objectives (De Falco et al. 2015, 2016). There exist other papers that refer to a multiobjective approach (Diosteanu et al. 2011; Cotfas 2011; Cotfas et al. 2011; Yon et al. 2012) or consider operating conditions and multiple constraints to derive personalized tours (Vansteenwegen et al. 2009b, 2011b; Souffriau et al. 2009, 2013; Huang and Bian 2009; Gavalas et al. 2012, 2017; Sylejmani et al. 2012).

Table 1 shows the comparison of the main features of some of the most complete PETGs suggesting walking itineraries (Vansteenwegen et al. 2011b; Cotfas et al. 2011; Gavalas et al. 2012, 2017; Brilhante et al. 2015). In the table *Y* signifies 'Yes' and *N* 'No' while NA indicates 'Not available' (the information is missing) and $\mathrm{NA}_p$ 'Not applicable' (the feature is not appropriate to the PETG in question). The table also reports the TTDP version solved by the underlying algorithm for the tour planner. The three optimization criteria in Cotfas et al. (2011) are the useful visiting time, the average score of the visited locations and the diversity of the itinerary. The table also contains in the first column the same information for our PETG.

It should be noted that the PETG relying on our optimizer is innovative because it includes is features that are missing in the other PETGs. Significant examples are the waiting time due to queues at the locations, the accessibility for disabled people

**Table 1** Presentation of the main features of some of the most recent and complete PETGs for walking itineraries in comparison with our PETG application

| | This paper | Vansteen-wegen et al. (2011b) | Gavalas et al. (2012) | Cotfas et al. (2011) | Brilhante et al. (2015) | Gavalas et al. (2017) |
|---|---|---|---|---|---|---|
| *MDays* | *Y* | *Y* | *Y* | N | *Y* | *Y* |
| *DSP* | *Y* | *Y* | *N* | $NA_p$ | *Y* | *Y* |
| *DFP* | *Y* | *Y* | *N* | $NA_p$ | *Y* | *Y* |
| *SFPNSame* | *Y* | *Y* | *N* | *Y* | *Y* | *Y* |
| *BR* | *Y* | *Y* | *Y* | *N* | *N* | *N* |
| *NBR* | ≥ 1 | 1 | ≥ 1 | 0 | 0 | 0 |
| *MTW* | *Y* | *Y* | *Y* | *N* | *Y* | *Y* |
| *CTW* | *Y* | *Y* | NA | *N* | *Y* | *Y* |
| *AVT* | *Y* | *Y* | *Y* | *Y* | *Y* | *Y* |
| *AWaitT* | *Y* | *N* | *N* | *N* | *N* | *N* |
| *AWalkT* | *Y* | *Y* | *Y* | *Y* | *Y* | *Y* |
| *PUser_Speed* | *Y* | *N* | *N* | *N* | *N* | *N* |
| *WeathFor* | *Y* | *N* | *Y* | *N* | *N* | *N* |
| *DisPeople* | *Y* | *N* | *N* | *N* | *N* | *N* |
| *NObj* | 5 | 1 | 1 | 3 | 1 | 1 |
| Problem | MO-TOPTW | Multiple TOP/TW | TTDP | TDOPTW | GMC | MTOPTW |

*MDays*: tour in multiple days, *DSP*: different start positions over different days, *DFP*: different finish positions over different days, *SFPNSame*: start and finish positions not necessarily the same, *BR*: break request, *NBR*: number of break requests, *MTW*: multiple time window for each *POI*, *CTW*: changing time window over days, *AVT*: average visiting time, *AWaitT*: average waiting time due to queues, *AWalkT*: average walking time between *POI*s, *PUser_Speed*: personalized user speed, *WeathFor*: weather forecast, *DisPeople*: disabled people, *NObj*: number of objectives

and the personalized moving speed of a tourist. Moreover, our guide is the only one that uses a trip planner algorithm capable of addressing this new and more complex extension of the TOPTW with five objectives and with a so many features. As a further issue, with the exception of one paper (Vansteenwegen et al. 2011b) that presents a questionnaire comprising six questions to assess the user satisfaction, our PETG is the only one that includes an in-depth evaluation at the end of the proposed itinerary.

## 3 Overview of our PETG application

The prototype of our PETG has been implemented as an App named 'Guida Elettronica per Napoli Antica (*GENnArí)*' devised in relation to the historical center of Naples within the project "Organization of Cultural Heritage for Smart Tourism and Real-Time Accessibility (ORCHESTRA)'.

The aim of GENnArí is to supply the tourist with a planned multiple-day itinerary by means of an interactive information exchange with the user. The flow diagram of the GENnArí application is schematized in Fig. 1. In particular, the personalized tour generator module, which represents the core of this guide, requires a definition and description of the user, the visit and the *POI* models (Gavalas et al. 2012).

The *user model* contains parameters or constraints such as personal interests, possible movement disabilities and average walking speed.

The *visit model* takes into account parameters like the amount of time available daily for visits, the number and duration of requested breaks, the daily points of departure and arrival, and environmental constraints.

The *POI model* requires for each site information such as waiting and visiting times, operating hours for each tour day, accessibility for disabled people, and whether it is an indoor or outdoor site.

The App installation requires a preventive authorization for specific functionalities and information on the mobile device such as: photo and video acquisition, geographic location to derive the geographic position, modification of the contents of the archive card, full access to internet and its connections so that the App can

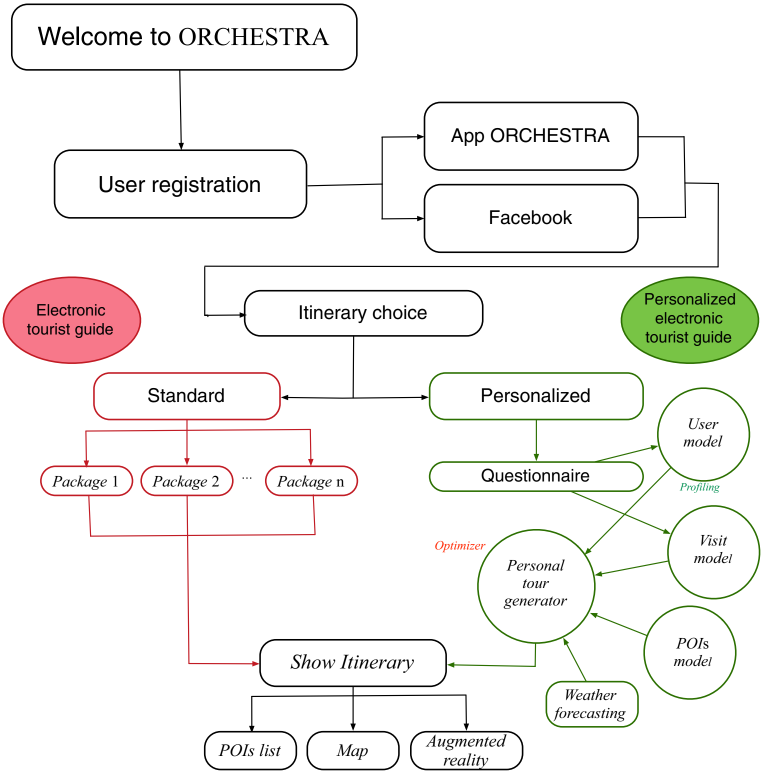



**Fig. 1** The activity flow diagram of the PETG

permit data download, and stand-by dis-activation that allows the disactivation of the geo-localization service.

Furthermore, we hypothesize that in full operation the site managers will have the responsibility to maintain and update the numerous items of information associated with the *POI model* and stored on a database server. Therefore, the App must be able to query this remote server, a feature which could delay the App functionality. This is the reason why, to reduce the access time and optimize the App start up, an automatic download of this information is effected. Moreover, an additional checking between the *POI* images on the server and those already contained in the respective device folder is carried out. The procedure simply downloads and stores the new images on the device memory so further reducing the App start time.

### 3.1 Registration

The user is encouraged to register her/his request for a personalized tour. The 'Registration' requires the selection of the registration modality, i.e., the Homepage of the App ORCHESTRA or the Facebook social network through which it is possible to derive user profile (Fig. 2). Once registered, the App maintains historical tracks of previously performed tours in case the user wishes to avoid the repetition of already visited *POI*s in future itineraries.

### 3.2 Itinerary choice

A registered user can choose between two alternative methods of creating the itinerary: standard and personalized. According to the selected option, the tourist has to supply a specific set of information to allow the device to arrange the requested tour.


**Fig. 2** Screenshots of the Homepage and of the registration modality

The standard route only requires the inputting of general information, while the personalized route also asks for personal information.

### 3.2.1 Standard route

A user who does not wish to provide too much information or has no specific requirements in terms of preferred *POI*s can request a standard tour, i.e, a generalized tour based on several alternatives of base and thematic packages relying on categories. We have considered a set of such categories, namely: amphitheaters, churches, museums, underground excavations, gardens, historical buildings, libraries, monuments and steps. An example on the specific theme of churches is shown in the left side of Fig. 3. These packages are arranged without taking into account any personal preferences and are simply based on the ratings assigned by experts in cultural heritage within the ORCHESTRA project with reference to the sights in general or to those associated with the selected theme. In fact, since each *POI* can be included in different categories, different scores as a function of the category of belonging can be assigned to each individual *POI*.

*Step 1* For each day of the tour, the visit model requires a declaration of the following items (see the right side of Fig. 3):

- start and end times for each day of the tour;
- number and duration of requested breaks;
- daily points of departure and arrival.

The constraint of available time is essential to propose a realizable tour. In fact, the addition of breaks required by the user is considered as an additional set of constraints, each treated as a 'soft *POI*' to be visited. This means that this special kind of *POI* should be included within the suggested tour, although the associated times

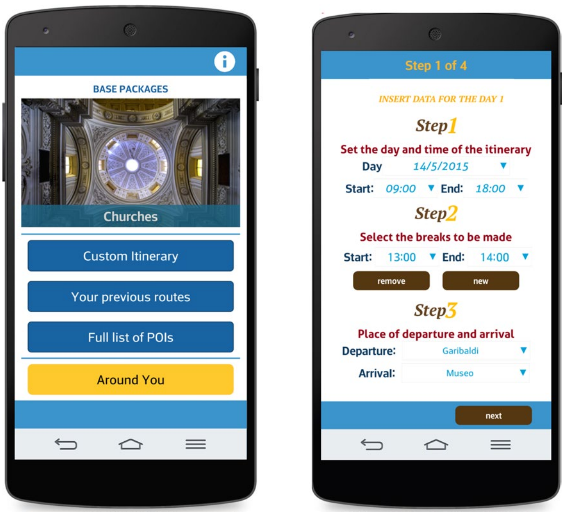



**Fig. 3** Screenshot of the itinerary choice and of *Step 1*

assigned by the tourist are considered as just approximate. Hence, differently from the hard constraints, such as for example the operating hours of the *POI*s, the breaks are treated as soft temporal constraints. There may be some flexibility in the organization of such breaks so that they may take place in appropriate neighborhoods at times demanded by the user, especially if such a flexibility permits a better satisfaction of preferences as a function of the opening times and of the number of visited *POI*s.

The start and arrival positions, internally represented by their GPS coordinates, i.e., longitude and latitude, are chosen by the user either from the list of tourist attractions, from the map shown on the mobile device, or from a set of 'access gates' to the old city center to important stops in the systems of local transportation. The possible 'access gates' are reported in Table 2.

#### 3.2.2 Personalized itinerary

In contrast, users interested in a tour tailored to their preferences must press the 'Custom itinerary' button. In addition to the information supplied during *Step 1*, they must provide information for the tour customization in successive steps: this information permits the definition of personal parameters/constraints to be taken into consideration for the user and the visit model. The App returns an itinerary even if not all the information requested in all the steps is given. In fact, the degree of personalization of such an itinerary is a function of the set of the information actually provided.

*Step 2* The user model requires a definition of the following information:

– possible movement disabilities;
– average walking speed.

In the same step, it is possible to set the weather forecast. The disability and weather forecast options can be enabled so that GENnArí takes into account limitations due to a disability and weather conditions in the planning of the tour.

**Table 2** The access gates for the old city center of Naples

| ID | Name of the access gates |
|---|---|
| 1 | Museo (M1/M2) |
| 2 | Dante (M1) |
| 3 | Montesanto (Cumana/Funicular of Montesanto/M2) |
| 4 | Toledo (M1) |
| 5 | Piazzetta Augusteo (Central Funicular) |
| 6 | Municipio (M1/M6/Port/Car Parking) |
| 7 | Università (M1) |
| 8 | Duomo (M1) |
| 9 | Garibaldi (M1/M2/Circumvesuviana/Car Parking) |

The system is able to take into account any possible walking speed, indicated by a real value ranging from a theoretical lower limit of 0.0 up to an upper limit of 1.0 m/s (3.6 km/h), which is a relatively high average speed for a tourist trip. The moving speed characterizing some typical tourist categories, such as a family with children (low average speed), a young person (high average speed), an old and disabled person (low average speed), can be logically connected to a specific subrange.

*Step 3* and *Step 4* Further requested information is related to:

- personal interests or preferences (*POI* category: e.g., museums, churches, or monuments) (*Step 3*),
- *POI*s desired to be included in the tour (*Step 4*).

The former deals with the parameters associated to the user model while the latter relates to the visit model. Screenshots relative to the second, third and fourth interactive steps are shown in Fig. 4.

The last piece of information needed to build a personalized itinerary is related to the *POI* model. This requires a definition of the following items for each *POI*:

- name;
- average waiting and visiting times;
- operating hours for each day of the tour;
- accessibility for disabled people;
- nature of the site, i.e., outdoors or indoors.

In GENnArí a list of 30 *POI*s within the selected area is taken into account. All the information associated with these *POI*s, included in the App, is outlined in Table 3.

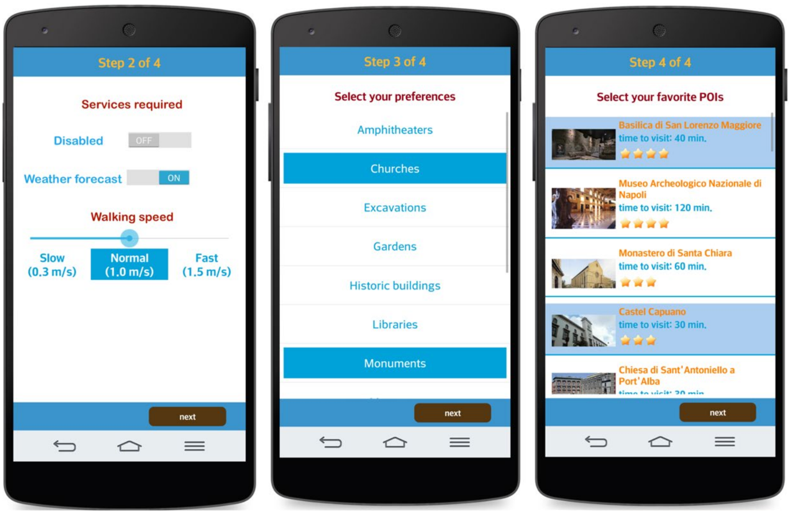



**Fig. 4** The screenshots of the *Step 2*, *Step 3* and *Step 4*

**Table 3** The *POI*s for the old city center of Naples

| ID | Name of the *POI* | Waiting time | Visiting time | Operating hours | Indoor | Disabled access |
|---|---|---|---|---|---|---|
| 1 | Basilica of San Lorenzo Maggiore | 0 | 40 | 9.30 am–5.30 pm | *Y* | *Y* |
| 2 | National Archaeological Museum | 10 | 120 | 9.00 pm–7.30 pm | *Y* | *Y* |
| 3 | Church of Santa Chiara | 10 | 60 | 7.00 am–1.00 pm, 4.30 pm–8.00 pm | *Y* | *Y* |
| 4 | Capuano Castle | 0 | 30 | 8.00 am–8.00 pm | *Y* | *Y* |
| 5 | Sant'Antoniello | 0 | 30 | 9.00 am–6.00 pm | *Y* | *Y* |
| 6 | Sansevero Chapel Museum | 5 | 30 | 9.00 am–6.30 pm | *Y* | *Y* |
| 7 | The Cathedral (Duomo) | 0 | 40 | 8.30 am–1.30 pm, 2.30 pm–8.00 pm | *Y* | *Y* |
| 8 | San Marcellino | 0 | 20 | 8.00 am–8.00 pm | *N* | *Y* |
| 9 | Roman Theater | 15 | 60 | 10.00 am–6.00 pm | *Y* | *N* |
| 10 | Church of San Gregorio Armeno | 0 | 30 | 9.30 am–5.00 pm | *Y* | *Y* |
| 11 | Church of Girolamini | 5 | 50 | 9.30 am–5.00 pm | *Y* | *Y* |
| 12 | Church of Santa Maria Donnaregina | 0 | 40 | 9.30 am–16.30 am | *Y* | *Y* |
| 13 | Basilica of San Paolo Maggiore | 0 | 30 | 10.00 am–6.00 pm | *Y* | *Y* |
| 14 | Diomede Carafa Palace | 0 | 60 | 10.00 am–1.30 pm | *Y* | *Y* |
| 15 | Filangieri Museum | 10 | 90 | 9.00 am–1.00 pm | *Y* | *Y* |
| 16 | Conservatory of San Pietro a Majella | 0 | 45 | 9.00 am–6.00 pm | *Y* | *Y* |
| 17 | Church of Gesù Nuovo | 0 | 40 | 6.30 am–1.00 pm, 4.00 pm–8.00 pm | *Y* | *Y* |
| 18 | Palazzo Venezia | 0 | 60 | 9.00 am–5.00 pm | *Y* | *Y* |
| 19 | Church of San Domenico Maggiore | 0 | 35 | 9.30 am–12.00 pm, 4.30 pm–7.00 pm | *Y* | *Y* |
| 20 | God Nile Statue | 0 | 10 | 12.00 am–12.00 am | *N* | *Y* |
| 21 | Cloister of Santa Chiara | 10 | 40 | 9.30 am–5.30 pm | *N* | *Y* |
| 22 | Piazza Bellini | 0 | 30 | 12.00 am–12.00 am | *N* | *Y* |
| 23 | Firrao Palace | 0 | 30 | 8.00 am–8.00 pm | *Y* | *Y* |

**Table 3** (continued)

| ID | Name of the *POI* | Waiting time | Visiting time | Operating hours | Indoor | Disa-bled access |
|---|---|---|---|---|---|---|
| 24 | Port'Alba | 0 | 20 | 12.00 am–12.00 am | *N* | *Y* |
| 25 | S. Aniello at Caponapoli (Cerere Temple) | 0 | 40 | 9.00 am–7.00 pm | *N* | *N* |
| 26 | Monte di Pietá | 0 | 25 | 9.00 am–6.00 pm | *N* | *Y* |
| 27 | Porta Capuana | 0 | 30 | 12.00 am–12.00 pm | *N* | *Y* |
| 28 | Royal Mineralogic Museum | 0 | 30 | 9.00 am–4.50 am | *Y* | *Y* |
| 29 | Anthropological and Zoological Museum | 5 | 60 | 9.00 am–16.50 am | *Y* | *Y* |
| 30 | Paleontology Museum | 5 | 40 | 9.00 am–16.50 am | *Y* | *Y* |

At this stage of the App development, the data associated with the *POI* model have been provided by experts in the cultural heritage. Naturally, these values have been used in the building of precomputed itineraries before the visit, at the planning stage. It is evident that the predicted values for average queueing and visiting times are not always reliable. Several methods, discussed in Sect. 7, are known to derive more reliable values. If these values should be made available by the *POI* managers, the App could perform a replanning of the initial programmed tour.

Furthermore, our App ‘GENnArí’ offers to the tourist two additional options, i.e., ‘Change itinerary’ and ‘I’m late’, which can both be activated after the start of the tour. The first option allows the setting of priorities among the five objectives considered by the planner (see Sect. 4) and the App re-computes the tour on the basis of these objectives. The second option can be invoked when the user has spent too much time at the sites so far visited. It outlines the list of *POI*s in the planned tour providing the user with the possibility of indicating those already seen. The App asks for other *POI*s to be visited and thus rearranges the tour.

### 3.2.3 Personal tour generator

All the information concerning the features of each *POI* is essential for the ‘Personal tour generator’ module, outlined in Fig. 1, to plan a personalized route and therefore it is stored in appropriate databases. A database contains the distances between the access gates and the *POI*s, measured by using the Google Maps automatic distance computation tool, and the routes between *POI*s.

The optimizer uses a procedure able to assign a numerical ‘score’ or ‘rating’ to each *POI* based on the personal interests and preferences of the tourist. Such a score assesses the *POI* according to its attractiveness and interest for that specific user. Notwithstanding recent advances in the design of tools for planning personalized tour recommendations, the problem of how to properly model preferences using online data sources and social media tools within the optimization process is till to be solved (Kotiloglu et al. 2017).

For our tour optimizer, we consider as available the values of the personal score for each *POI* during the planning of the tour. Specifically, the user can indicate preferred *POI*s or declare an interest in particular categories. In the former case, the maximum score value is associated with the selected *POI*s. In the latter, the values are obtained by multiplying the scores assigned by the experts to each single *POI*, as a function of the belonging category, by a factor explicitly declared by the user in a fixed range when choosing that category. If the user does not express any preferences, the used values are those established by the experts. More accurate values can be automatically obtained by using techniques such as, e. g., those reported in Sect. 7.

This information is also coupled with the weather forecast module. This is employed to take into account the weather conditions so as to suggest, as far as possible, routes which include the visits to indoor *POI*s in the case of rain, snow, strong winds or excessive heat. Currently, open-access sites are able to supply hourly weather forecasting, yet such forecasting is extremely complex. Hence, as an example, if rain is predicted at a given time, it is reasonable to assume that the probability of rain is enclosed within a larger time slot which includes the above-mentioned time. A practical hypothesis to define the time slot of the adverse weather condition is to presume that the rain can occur within one hour before or after the indicated rain interval. Therefore to predict the weather conditions, it is considered that a granularity equal to three hours is reasonable. Consequently, the table containing the weather forecasting for the old center of Naples is subdivided into 3-h time slots. In particular, once a user requests a personalized tour, a weather table will be specifically created for her/him. This will include a number of items equal to that of the 3-h time slots related to the days in which the tour falls. Naturally, weather forecasting is addressed dynamically during the evaluation of the solution encoding the tour, and suggests the exclusion of an outdoor *POI* from the tour under construction if and only if one of the adverse conditions is forecast at the time in which the outdoor *POI* should be visited. Specifically, during the planning of the tour, the algorithm keeps track of the 'current time' needed for the visit of all the *POI*s already included. Whenever our tool has to include a new *POI* in the itinerary that it is generating, it takes into account the next *POI* in the sequential order in the solution in question, the 'current time' based on the *POI*s already inspected and on the breaks that have already occurred, the weather forecasting table and the 'Indoors' attribute of the *POI* under consideration. If at the 'current time' the weather forecasting predicts bad weather conditions and if the *POI* under consideration is outdoors, then that *POI* is not inserted in the tour to be proposed, and the subsequent one in the solution is considered for possible inclusion instead.

Within the ORCHESTRA project the weather forecast is obtained from the weather service managed by the 'Centro Campano per il Monitoraggio e la Modellistica Marina ed Atmosferica' (CCMMMA) (Campanian Center for Marine and Atmospheric Monitoring and Modeling). The information is available through the official web page of the Center: http://meteo.uniparthenope.it.

#### 3.2.4 Show itinerary

There are two tour presentation modalities, namely textual and visualization on a city map. In the first case, the generated tour is shown as a list of *POI*s chronologically ordered, containing information on each *POI* to visit, and on the arrival, waiting and visiting times. In the second case, the itinerary is presented on a geographical map which shows the route by highlighting the *POI*s to visit. This visualization is obtained by using the Google Maps API, a web service that allows users to find the route between two or more points. The final output is a polyline whose nodes represent all the *POI*s included in the itinerary and the arcs their respective connections. The zooming ability of the mobile device allows users to obtain a detailed visualization of the suggested tour as well as the streets to walk along. Examples of these two different modalities are reported in Fig. 5: the first two screenshots are related to the textual modality while the third one shows the visualization on the city map.

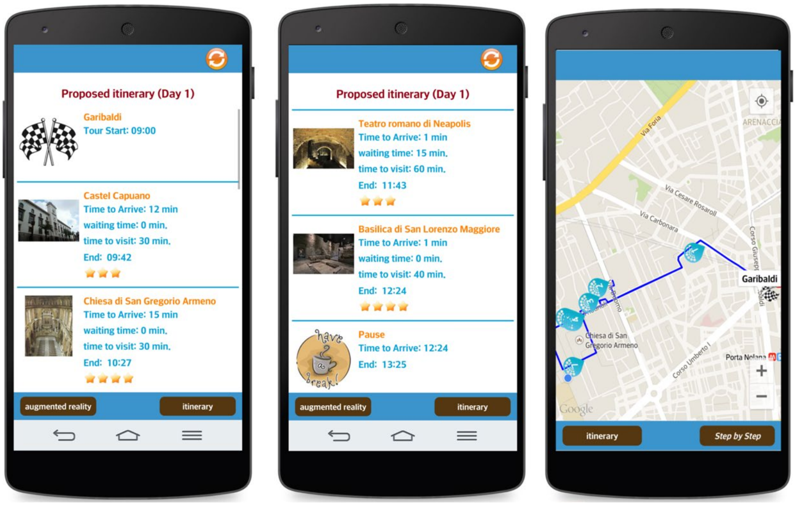



Fig. 5 Screenshots of the textual and map modalities

### 3.2.5 Your previous routes

The App maintains a data file which keeps track of the previous tours made by the tourist. This information is useful in the case of a new tour being planned by a user who has previously visited the same area. This knowledge of previous tours avoids the possibility of a route containing already visited *POI*s being proposed, unless the user expressly requests to visit again some of these *POI*s.

### 3.2.6 Information about *POIs*

The information about each individual *POI* is subdivided into three principal sections that are visualized through their respective buttons:

- a description that shows a brief text about the *POI*;
- a presentation of the opening hours and contacts related to the *POI*;
- a map that visualizes the *POI* position on a map or in augmented reality.

Screenshots of the provided *POI* information are shown in Fig. 6.

## 3.3 Additional services

### 3.3.1 Augmented reality visualization

The App ‘GENnArí’ also allows the exploration of a tourist area by means of augmented reality. This provides an intermediate layer between the real environment and the user by adding virtual content to the visual experience. The real layer is acquired by the smartphone camera while the information layer is derived from data

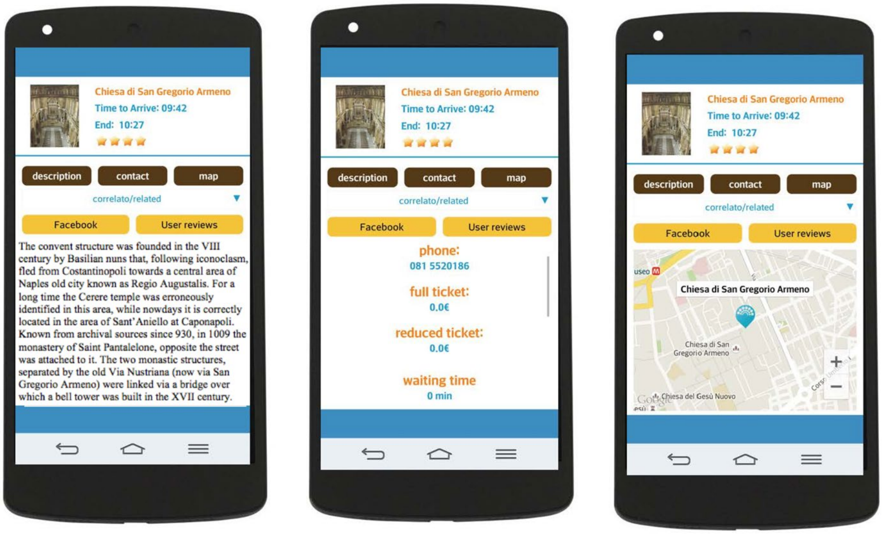



**Fig. 6** Screenshots of the *POI* information

available in the App. The outdoor visualization of augmented reality is typically based on information about the position of the user and envisages an integration of information. The development tool used is SDK Wikitude (Wik 2017). Wikitude is a mobile application for augmented reality that exploits GPS, and the compass and accelerometer of the device to obtain the position. The modality is activated by pressing a specific button. The result is visualized by means of a marker on the display, with the possibility of executing interactive operations such as navigation, information and 3D visualization. The information visualized on the smartphone screen is the position of the *POI*s and of the access gates with respect to the current user position. The number attached to each *POI* indicates the visit order. Examples of the augmented reality modality are reported in the Fig. 7: the 'access gates' are indicated in red while the *POI*s are shown in green.

### 3.3.2 Visualization on smartwatches

The App must allow the user to concentrate on the visit rather than on the interaction with the device, or otherwise, the whole user experience is limited to a man–machine interface. Recognizing this has pushed us to design the App integrated with a wearable devices to guarantee a free-hand experience. We have evaluated different approaches by concentrating our efforts on wearable device linked via bluetooth to the mobile device. Within the ORCHESTRA project a GUI on an Android Wear smartwatch has been implemented. Therefore, there is an additional visualization modality for use with a smartwatch (see Fig. 8). This represents an alternative modality which avoids the necessity of having a hand engaged with the smartphone. When the user is in the area of a *POI*, the application automatically reports its position through brief notifications on the smartwatch. Although not personalized, this

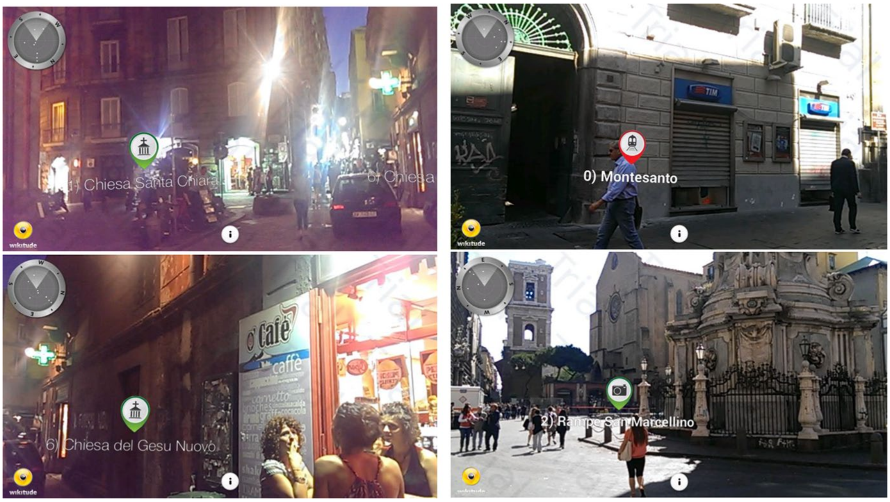
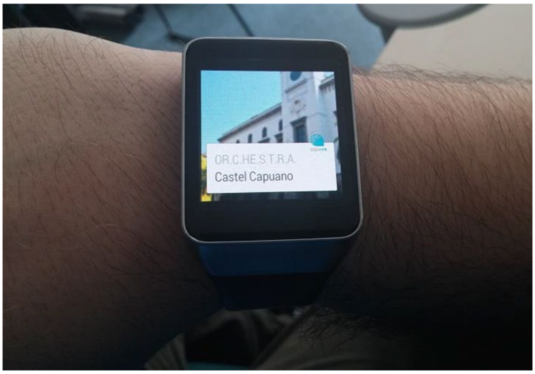



**Fig. 7** Screenshots of the augmented reality modality

**Fig. 8** Screenshot of the smartwatch




acoustic and visual information, delivered dynamically during the tour, allows the provision of contextualized content and allows the tourist to enjoy the visit without continuously having a look at the smartphone screen.

### 3.3.3 Around you

The implementation of the 'Around you' functionality exploits the API of Google Maps. This functionality allows visualization of a set of services around the user, meaning that she/he can select any of them in which she/he is interested. An additional facility consists in the generation of notifications of services through the application of a dynamic distance threshold of at most 50 m from the current position. Such a facility is disabled in the default configuration to avoid the possibility of these notifications becomin too frequent and, thus, irritating users, who, otherwise,

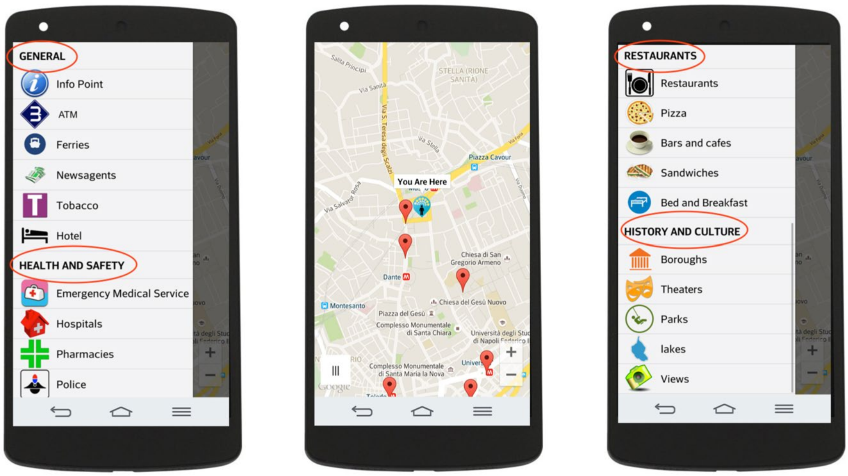



**Fig. 9** Screenshots of the ‘Around you’ functionalities

would have to explicitly configure this deactivation. A user who has explicitly activated this facility will automatically receive notifications about different categories such as *POI*s, restaurants, bars, pharmacies, information points., etc. (see Fig. 9). As in the case of the augmented reality facility, the displayed categories are not personalized but simply help the user to take decisions. For example, the ‘Around you’ functionality can also suggest ‘soft points’ where the tourists can enjoy their planned breaks during the tour.

## 4 The evolutionary optimizer

The personalized tour is planned by the evolutionary optimizer that is the App core. The logical architecture illustrating the interactions between the optimizer computational unit and the input/output modules is reported in Fig. 10. This optimizer interacts with the previously described input modules to collect information about the visit, the user and the *POI* models, through a profiling phase, and with an output module to provide the tourist with the customized multiple-day itinerary. The time required for the retrieval of this itinerary depends on the computational power of the mobile device used. Hence, it can range from just a few seconds on the most advanced smartphones and tables, up to about 60 seconds on old and low-performing mobile devices.

The optimizer plans a personalized itinerary with the aim of achieving the following objectives as effectively as possible:

– to maximize the attractiveness of the proposed tour, starting on each day at a specific place and ending at a given destination, in such a way that in the

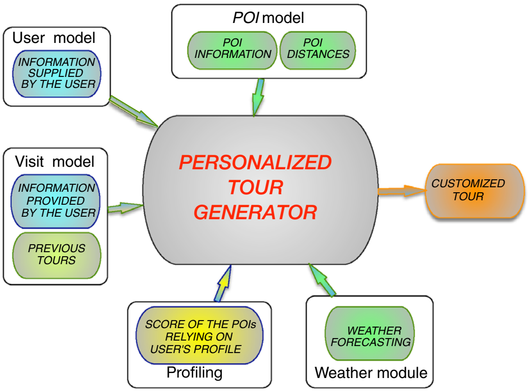



**Fig. 10** The logical architecture of the personalized tour generator

itinerary no *POI* is selected more than once and no *POI* incompatible with disability is identified;

- to visit as many of the *POI*s expressly requested by the user as possible
- to visit as many *POI*s as possible in addition to those expressly requested by the user;
- to minimize the distance covered;
- to complete the tour within the time limit established by the user respecting the following constraints: (i) opening and closing times of the *POI*s, (ii) waiting times and visit durations, (iii) break times, (iv) accessibility, (v) weather forecasts and (vi) walking speed.

The mathematical formalization of all the objectives addressed by our optimizer is outlined in the following.

- *Score of the tour* Denoting with a real value $s(i)$ the score of a generic $POI\ i$ for a specific user and with $POI_{intour}$ the number of *POI*s included in the suggested tour, the score of the complete path *tour* is equal to:

$$\Phi_1(tour) = \frac{1}{POI_{intour}} \sum_{i=1}^{POI_{intour}} s(i) \tag{4.1}$$

  This objective is to achieve the maximum value.
- *Number of POIs expressly requested by the user and effectively included in the suggested tour* Indicating with $POI_{req}$ the number of *POI*s that the user requests to be included in the tour, an ideal tour will be one which contains all the $POI_{req}$. Nonetheless, due to user and environmental constraints, the tour proposed by the optimizer could include a smaller number of requested *POI*s, named $POI_{act}$, lower than or equal to $POI_{req}$. This objective is to be minimized and is expressed in a formula as:

$$\Phi_2(tour) = POI_{req} - POI_{act}$$

- *Number of total POIs included in the tour* The user is interested in visiting as many *POI*s as possible in the available time for the visit. Therefore, the optimizer has to suggest a tour with the largest number of *POI*s best matching the user preferences. So we can define the following objective to be maximized:

$$\Phi_3(tour) = POI_{intour}$$

- *Tour length* Another important issue to take into account is the total distance covered during the tour. This must not be excessively long, especially for some categories of potential users, such as elderly people, families with children, or disabled people. It should be noted that the classic Traveling Salesman Problem, which to find a path only considers distances, cannot be used in this situation. In fact, for a PETG, in addition to the spatial coordinates, the temporal aspect, represented for example by the opening and closing times of the *POI*s, is also to be taken into account. However, it is crucial also to consider a spatial objective aimed at minimizing the length of the suggested tour $\boldsymbol{\Phi}_4(tour)$ evaluated as:

$$\boldsymbol{\Phi}_4(tour) = d_{in} + \sum_{i=1}^{POI_{intour}} d_{i,i+1} + d_{fin}$$

  where $d_{in}$ indicates the distance between the start position declared by the user and the first *POI* in the tour under consideration, $d_{i,i+1}$ is the distance between a generic *POI* $i$ and the successive *POI* $(i + 1)$, and $d_{fin}$ represents the distance between the last *POI* in the tour and the finish point stated by the user.
- *Total duration of the tour* It is assumed that the user wishes to spend as much time as possible during the visits of the attractions and also wishes to spend time for breaks (rests, lunch, shopping, etc.). The total time of the tour is evaluated as:

$$\boldsymbol{\Phi}_5(tour) = \sum_{j=1}^{POI_{intour}} T_{v_j} + \sum_{k=1}^{N_p} t_p(k) + t_{arr}$$

  where

  - $T_{v_j}$ is the sum of:
    - $t_{i,j}$ that denotes the time needed to move from a *POI* $i$ to $j$, where $j$ indicates the *POI* successive to $i$ in the tour under consideration (in the case of the first *POI* it identifies the time needed to reach the first *POI* from the start point selected by the user);
    - $t_{w_j}$ which represents the waiting time needed to enter the *POI* $j$, due to queues, or other constraints;
    - $t_{v_j}$ that indicates the visiting time taken at the *POI* $j$;
  - $t_{arr}$ denotes the time needed to reach the finish point selected by the user from the last visited *POI*;
  - $t_p(k)$ represents the duration of the generic break $k$ requested by the user (let $N_p$ be the total number of such breaks).

This quantity is to be maximized while respecting the constraint $\Phi_5(tour) \leqslant max_{t_{tour}}$, where $max_{t_{tour}}$ indicates the time limit stated by the user to complete the tour.

Besides the available time, the opening and closing times of the sites, the waiting and visiting times, and the breaks all require a further temporal limitation. In fact, as already stated in Sect. 3.2.1, the requested breaks are treated as soft temporal constraints. Specifically, as is the case in relation to the weather forecasting, during the tour construction, we monitor the 'current_time' needed for the visit of all the *POI*s already included in the tour in question. Before adding an additional *POI*, we check whether there is sufficient time to satisfy the request for a break.

The suggested route must also respect other constraints such as: (i) accessibility to the *POI*s according to specific categories of users (in the case of tours for disabled people, *POI*s with limited access are to be excluded); (ii) possibility of visiting outdoor *POI*s as a function of weather conditions (in the case of rain, snow, strong winds or excessive heat, outdoor *POI*s are to be excluded from the routes, unless explicitly requested by the user).

It is worth highlighting that the above-listed objectives can be conflicting. For example, visiting all the requested *POI*s could involve an excessively long distance, and the inclusion of some of these *POI*s may involve the exclusion of others that would increase the score of the proposed tour.

To address the MO-TOPTW with the above-mentioned five objectives, we use a multiobjective evolutionary algorithm relying on Pareto dominance and a Pareto front. A comprehensive description of this topic can be found in Deb (2001) and Coello et al. (2007). The pseudo-code description of the algorithm is schematically reported in Algorithm 1.

**Algorithm 1** Pseudo-code of the optimizer

```
Require
Information provided by the user on dates, available tour times, preferences and constraints
Information on features and distances, weather forecasting and previous tours
begin
  for i=1 to NPOP (Population size of the evolutionary algorithm)do
    initialize randomly a tour t_i
    evaluate the tour t_i
  compute the initial Pareto front
  for i=1 to g_max (Maximum generation number of the evolutionary algorithm) do
    for j=1 to NPOP do
      choose randomly two tours t_1 and t_2 on the Pareto front
      recombine these two tours to generate a child t_c
      mutate t_c by means of evolutionary operators
      compare t_c with the i-th tour in the current population
      save the best (i.e., the non dominated) between these two tours in the new population
    evaluate the new Pareto front
  select the 'best' tour on the Pareto front
end
Output the proposed tour
```

Each population solution represents a potential multiple-day tour and is encoded by a vector of integer values with a dimension equal to the number of *POI*s. Each integer identifies a *POI* and is only present once in each solution. As an example, a solution with ten *POI*s is shown in Fig. 11. Such a tour, starting from the position selected by the user (not expressly contained in this encoding), suggests reaching the *POI* in the first cell at the left side of the vector (6 in the figure), then moving forward to visit the *POI* in the second cell (10 in the figure), and continuing in this fashion. In the algorithm, the 'best' tour on the Pareto front is the one with the lowest euclidean distance from the theoretically optimal point. A more in-depth description of this evolutionary optimizer and implementation details is reported in De Falco et al. (2016).

## 4.1 Tour examples

To better illustrate the planning of the walking tours in the historical center of Naples that our App is able to propose, two standard types of tourists are reported in the following subsections, along with a discussion of the capability of the App to satisfy the tourists' multiple objectives and constraints. Readers interested either in examining other real walking itineraries, proposed by GEN-nArí in accordance with different user interests and conditions, or in a numerical comparison of the results of our multi-objective EA with those of simpler mono-objective EAs, are referred to our papers (De Falco et al. 2015, 2016).

### 4.1.1 First example

As a first example, we will consider the case of a tourist arriving in the morning at Piazza Municipio disembarking from a cruise ship in the nearby city harbour, and having 1 day (9.30 am–8.00 pm) to take a tour before leaving the city in the evening. Therefore, the arrival and the departure are considered here as taking place at Municipio (M1) metro station. Such a tourist wishes to visit five *POI*s in Table 3 (the National Archaeological Museum (id #2), the Church of Santa Chiara (id #3), the Basilica of San Lorenzo Maggiore (id #1), the Cathedral (id #7) and the Sansevero Chapel Museum (id #6)), and aims take a long break for lunch and another shorter break in the afternoon. The tourist also declares an interest in museums, churches and historical buildings. Therefore, the input information provided by the user is the following:

- Day of tour: 13/09/2019
- average moving speed: 1m/s
- disabled access request: no
- number of days available for the tour: 1
- number of *POI*s that the user states to be enclosed in the visit: 5

**Fig. 11** Example of a tour encoding

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 6 | 10 | 4 | 5 | 7 | 8 | 3 | 2 | 9 | 1 |

– identification codes of these *POI*s: 2, 3, 1, 7 and 6
  – start and finish times: 9.30 am–8.00 pm
  – start and finish positions: Municipio (M1)
  – number of programmed rests: 2
    - start and finish times of break 1: 12.30 am–2.00 pm
    - start and finish times of break 2: 4.30 pm–5.00 pm.

The best solution found on the Pareto front is the tour shown in Table 4 in which the waiting and visiting times are indicated in minutes.

This table is compact and easy to interpret: the tour starts at 9:30 am from the departure position chosen by the user; the *POI* with identifier #3 (the Church of Santa Chiara) is proposed as the first to visit. The walking time to reach this *POI* is estimated at 17 min, a waiting time of 10 min for the entrance is foreseen due to the queu of visitors, while the average visiting time is 60 min. The visit terminates at 10.57 am. The tour proceeds to *POI* #18 (the Palazzo Venezia) reachable in 4 min. This visit ends at 12.01 am. The next visited *POI* is that with identifier #6 (Sansevero Chapel Museum) after which the first break is programmed. Such a break finishes at 2.11 am (note that this was requested from 12.30 am to 2.00 pm, so it is very close to the user's requirements). The tour continues with the following programmed sites, interspersed by another break planned in the afternoon, until the arrival at the selected destination at Municipio (M1) station/Naples port occurs at 7.44 pm.

**Table 4** The output of the optimizer in the first example

| The proposed tour | | | | | | | |
|---|---|---|---|---|---|---|---|
| *POI* (break) | Start time | Walking time | Waiting time | Visiting time | Visit end | Break begin | Break end |
| Start | 9.30 am | | | | | | |
| 3 | | 17 | 10 | 60 | 10.57 am | | |
| 18 | | 4 | 0 | 60 | 12.01 pm | | |
| 6 | | 5 | 5 | 30 | 12.41 pm | | |
| (Break1) | | | 90 | | | 12.41 pm | 2.11 pm |
| 1 | | 7 | 0 | 40 | 2.58 pm | | |
| 7 | | 4 | 0 | 40 | 3.42 pm | | |
| (Break2) | | | 30 | | | 3.42 pm | 4.12 pm |
| 2 | | 19 | 10 | 120 | 6.41 pm | | |
| 23 | | 7 | 0 | 30 | 7.18 pm | | |
| Arrival | | 26 | | | 7.44 pm | | |
| Summary | | | | | | | |
| Requested *POI*s | | | 5 | Out of | 5 | | |
| Visited *POI*s | | | 7 | Out of | 30 | | |
| Total tour time | | | 614 | Out of | 630 | | |

The remaining part of the table contains a brief summary showing that the suggested tour allows the user to visit all the requested *POI*s and a total of 7 out of the 30 total available *POI*s, involves a total tour time of 614 min out of the 630 min available so avoiding waste of time. Moreover, the results in the table show that the two breaks have been set at times close to those requested by the user, the differences being of 11 and 48 min, respectively.

From the above findings, it is evident that the tour is planned according to the user preferences and information. In fact, the programmed itinerary includes visits to all the sites explicitly requested. It also includes two historical buildings matching the preferences stated by the user in her/his declaration of interest. This allows the maximization of the personal score of the itinerary. Furthermore, the total duration time of the trip is very close to the total available time, so allowing the user to spend as much time as possible looking at the attractions, consequently enhancing her/his level of satisfaction.

#### 4.1.2 Second example

As a second example a young user is considered who arrives at Garibaldi central station, directly by train or by Alibus express shuttle from the airport, and has two consecutive days available for the visit. Such a tourist is traveling with a little luggage so that she/he is able to begin immediately the tour from Garibaldi before returning to a bed and breakfast already booked in the nearby Piazza Dante in the evening of the first day. The starting point for the second day of the tour is Piazza Dante and the finish position is Piazza Museo where the tourist can use the metro to reach Garibaldi station where a train or the Alibus to the airport can be taken. She/he requests two breaks each day and wishes to visit seven *POI*s (the National Archaeological Museum (id #2), the Sansevero Chapel Museum (id #6), the Roman Theater (id #9), the Filangieri Museum (id #15), the Church of Gesù Nuovo (id #17), the Church of San Domenico Maggiore (id #19) and Piazza Bellini (id #22)). The user also declares an interest in museums, churches, historical buildings, monuments, amphitheaters and steps.

The input information supplied by the user is the following:

- Day of tour beginning: 16/06/2019
- average moving speed: 1 m/s
- disabled access request: no
- number of days available for the tour: 2
- number of *POI*s that the user requests to be in the visit: 7
- identification codes of these *POI*s: 2, 6, 9, 15, 17, 19 and 22
- For the first day of the tour:
  - start and arrival times: 10.00 am–8.00 pm
  - start position: Garibaldi (M1)
  - finish position: Dante (M1)
  - number of programmed breaks: 2
    - start and finish times of break 1: 12.00 am–1.30 am
    - start and finish times of break 2: 4.00 pm–5.00 pm

– For the second day of the tour:

  – start and finish times: 9.00 am–6.30 pm
  – start position: Dante (M1)
  – finish position: Museo (M1/M2)
  – number of programmed breaks: 2

    • start and finish times of break 1: 12.00 am–1.00 am
    • start and finish times of break 2: 4.00 pm–4.30 pm.

Table 5 can be interpreted using the same method described for Table 4. The summary part of the table reports a brief compendium and confirms that the suggested

**Table 5** The output of the optimizer for the second example

| *POI* (break) | Start time | Walking time | Waiting time | Visiting time | Visit end | Break begin | Break end |
|---|---|---|---|---|---|---|---|
| The tour proposed for the first day | | | | | | | |
| Start | 10.00 am | | | | | | |
| 15 | | 17 | 10 | 90 | 11.57 am | | |
| (Break 1) | | | 90 | | | 11.57 am | 1.27 pm |
| 9 | | 9 | 15 | 60 | 2.51 pm | | |
| 13 | | 1 | 0 | 30 | 3.22 pm | | |
| (Break 2) | | | 60 | | | 3.22 pm | 4.22 pm |
| 2 | | 17 | 10 | 120 | 6.49 pm | | |
| 22 | | 9 | 0 | 30 | 7.28 pm | | |
| 24 | | 1 | 0 | 20 | 7.49 pm | | |
| Arrival | | 4 | | | 7.53 pm | | |
| The tour proposed for the second day | | | | | | | |
| Start | 9.00 am | | | | | | |
| 6 | | 10 | 5 | 30 | 9.45 am | | |
| 28 | | 6 | 10 | 90 | 11.31 am | | |
| (Break 1) | | | 60 | | | 11.31 pm | 12.31 pm |
| 17 | | 9 | 0 | 40 | 1.20 pm | | |
| 19 | | 7 | 0 | 35 | 2.02 pm | | |
| 10 | | 8 | 0 | 30 | 2.40 pm | | |
| 11 | | 3 | 5 | 50 | 3.38 pm | | |
| (Break 2) | | | 30 | | | 3.38 pm | 4.08 pm |
| 16 | | 11 | 0 | 45 | 5.04 pm | | |
| 5 | | 2 | 0 | 30 | 5.36 pm | | |
| 25 | | 6 | 0 | 40 | 6.22 pm | | |
| Arrival | | 3 | | | 6.25 pm | | |
| Summary | | | | | | | |
| Required *POI*s | | | 7 | Out of | 7 | | |
| Visited *POI*s | | | 15 | Out of | 30 | | |
| Total tour time | | | 1158 | Out of | 1170 | | |

tour allows the tourist to visit all the requested *POI*s and a total of 15 out of the 30 total *POI*s, the tour taking a total time of 1158 min out of the 1170 min available so avoiding any waste of time.

As in the previous case, it should be noted that the two breaks have been set at times close to those requested by the user, the differences being of 3 and 38 min, and 29 and 22 min for the first and second day respectively. It is evident that the itinerary has been planned according to the preferences and information derived from the user profile. In fact, it includes, besides all the requested sites, churches and monuments that match the user preferences and allow a maximization of the personal tour score. Moreover, the total duration time of the tour is very close to the total available time so allowing the tourist to spend most of her/his time enjoying the visit.

## 5 Methodology used for the evaluation

GENnArí is a mobile App usable on all mobile devices with Android or iOS operating systems provided with network functionalities. This App prototype was evaluated publicly during a number of demonstration days where several mobile devices were handed to a group of volunteer participants. The skills required were not technical so as to permit a large-scale voluntary experimentation. The number of participants was 40. The volunteers belonged to four different age groups; in detail five were less than 18 years old, fifteen between 18 and 35 years, fifteen between 36 and 55 years and the remaining five over 55. 18 were females and 22 males. Moreover, they were asked to rate their familiarity with smart devices choosing three levels: low, medium and high. 5, 20 and 15 participants declared a low, medium and high experience respectively. Of course, none of them had previously used GENnArí. During the experimentation, the Android and iOS versions of the App were downloadable from the official site of the project (http://www.orchestrasmartnapoli.it/site1/eventi-orchestra.php). At the moment, due to legal restrictions imposed by this project, the App cannot be redistributed even free of charge.

The experimentation, carried out autonomously by the participants, consisted in the downloading of the App on provided or personal mobile devices and in its real-time experimentation around the historical center of Naples. The suggested itineraries were examined by the users exploiting the different visualization modalities offered by the PETG. During this activity the participants could report any bugs, suggestions and comments by interacting through an apposite screen. At the end of the trial, they were also required to complete an electronic evaluation questionnaire to provide feedback on the App. This data collection had a twofold objective: first, to validate the implemented App and, secondly, to test the Human–Computer Interface (HCI) of usability. The results of this experimental phase have been analyzed as a function of: (i) the number of registrations completed by the participants who downloaded and used the App; (ii) the number of evaluation questionnaires completed; and (iii) the scores, comments and reviews in the questionnaires collected during the experimentation.

A qualitative and synthetic analysis, by means of descriptive statistics tools, of the results obtained from an analysis of the evaluation questionnaires completed by the volunteers immediately after the trial of the prototype is presented. On the basis of the type of questionnaire, the responses that the volunteers gave to the items have been either analyzed separately, or examined in combination with related items so creating an aggregated score for that item group. The related items are those that should be considered as a whole to evaluate the same aspect of the user interactive experience.

To test the HCI of the prototype, particular attention has been devoted to the usability of the application and its GUIs, since this represents a crucial point in the development of a software system (Nielsen 1993; Preece et al. 2002; Dix et al. 2004). The International Organization for Standardization (ISO) defines the usability according to two different standards, namely ISO/IEC 9126 and ISO 924. Taking into account these two definitions, the usability is given by a series of factors that affect the user experience in the interaction with the system under examination.

Key factors considered for the evaluation of such an interaction are:

- *intuitiveness and ease of learning* how much time and effort are required by a user to learn the execution of an activity;
- *execution velocity* how much time a user needs to carry out an activity;
- *error frequency* how many and which errors a user tends to commit in executing the activity and how much harm such errors can cause;
- *persistence in time* how much time the user can continue using the application before forgetting the procedure;
- *personal satisfaction* comfort and acceptance (can the use modality of the tool affect the whole working method of the participant in the activity execution?).

The participants in the pilot study were asked to assess the adequacy and completeness of the App functionalities and facilities in providing personalized tours in realistic cases.

## 5.1 Experimentation

During the experimentation, the participants performed a series of prototype tests followed by the compilation of questionnaires extracted from several studies. The questionnaires were employed to measure the user experience on the basis of a Likert scale (Betram 2009). The questionnaires consisted of a certain number of statements (named items) to allow the user to express a positive or negative judgement with regard to a specific subject. An addition of these judgements was performed in order to indicate in a reasonably precise way the user's attitude. For each item an agree/disagree scale on 5 or 7 levels depending on the proposed query was presented. The participants were requested to assign a score on such scales. The intermediate value 3, or 4, corresponded to a neutral response.

The reason for the choice of the Likert scale is explained by the fact that it measures attitudes and behaviors by using a series of options ranging between two extremes. The fortune of such a technique is due to a series of advantages in its application. The style chosen for the response registration is very simple. Therefore the user will have little difficulty in the understanding of the questions and in the registration of the corresponding responses. Moreover, differently from a scale with two contrasting alternatives, the Likert scale permits a better articulation the personal opinion by allowing the user to select alternative responses. Another advantage derives from the easy orderability of the categories of the responses themselves over a continuum, thus enabling the responses given by the participants to be also ordered in a readily comprehensible manner.

Two different test questionnaires with scores based on the Likert scale were used for a subjective evaluation of the aspects related to the user interaction, namely:

- Usefulness, Satisfaction and Ease of Use Questionnaire (USE) (Lund 2001);
- User Experience Questionnaire (UEQ) (Laugwitz and T. Held 2008).

#### 5.1.1 Usefulness, Satisfaction and Ease of Use Questionnaire (USE)

The USE questionnaire was compiled by utilizing a five-level Likert scale with scores ranging from the lowest 1 corresponding to ‘strongly disagree’ to the score 5 for ‘strongly agree’. The lowest scores always corresponds to a negative evaluation while the highest ones to a positive judgement. We chose 30 items (see Table 6). These items were subdivided into four groups each exploring different aspects:

- *utility perspectives* the related items tend to quantify the user’s assessment of the application’s utility;
- *ease of use* the user’s ability to complete the task is considered;
- *ease of learning* the capability of the interface elements to be auto-explicative;
- *satisfaction* items associated with the general satisfaction of the user.

All the items of the questionnaire are characterized by the fact that they are positive statements. This aspect could cause a slight distortion towards the positive denoted as ‘acquiescence bias’. This distortion can be avoided by balancing the positive statement (‘I found the interface easy to use’) with a negative one (‘I found the interface difficult to use’). It should be noted that questionnaires with balanced items can suffer from other distortion modalities, such as for example reliability, since they can present different results if executed at different moments.

#### 5.1.2 The User Experience Questionnaire (UEQ)

The UEQ allows a rapid evaluation of the ‘user experience’ of interactive products. The questionnaire format supports the users in expressing in an immediate way the feelings, impressions and attitudes which emerge during the product use. The scales are related to a global impression of the user experience, namely they measure both the classic usability issues (efficiency, clarity, and reliability) and user experience issues (originality, and stimulation). So, adjectives such as

**Table 6** The USE questionnaire

| Item | Question | Strongly disagree | Disagree | Neither agree nor disagree | Agree | Strongly agree |
|---|---|---|---|---|---|---|
| *Usefulness* | | | | | | |
| 1 | It helps to enhance the benefits of the cultural heritage | o | o | o | o | o |
| 2 | It helps to satisfy my curiosity about the cultural heritage | o | o | o | o | o |
| 3 | It is useful | o | o | o | o | o |
| 4 | It permits a direct control on the cultural heritage information | o | o | o | o | o |
| 5 | It allows reaching my objectives in a simple manner | o | o | o | o | o |
| 6 | It permits saving time | o | o | o | o | o |
| 7 | It encounters my needs | o | o | o | o | o |
| 8 | It operates as I expect | o | o | o | o | o |
| *Ease of use* | | | | | | |
| 9 | The use is intuitive | o | o | o | o | o |
| 10 | It is simple to use | o | o | o | o | o |
| 11 | It is endowed with a simple interface | o | o | o | o | o |
| 12 | It requires the least number of steps to reach my objectives | o | o | o | o | o |
| 13 | It is flexible | o | o | o | o | o |
| 14 | It is usable without effort | o | o | o | o | o |
| 15 | It is usable without reading instructions | o | o | o | o | o |
| 16 | No incongruity is noticed during the use | o | o | o | o | o |
| 17 | Both casual and expert users can like it | o | o | o | o | o |
| 18 | I can correct the mistakes in an easy and fast manner | o | o | o | o | o |
| 19 | I can use it every time with success | o | o | o | o | o |

**Table 6** (continued)

| Item | Question | Strongly disagree | Disagree | Neither agree nor disagree | Agree | Strongly agree |
|---|---|---|---|---|---|---|
| *Ease of learning* | | | | | | |
| 20 | I learned immediately to use it | o | o | o | o | o |
| 21 | I easily remember its use | o | o | o | o | o |
| 22 | The use is easily comprehensible | o | o | o | o | o |
| 23 | I have been immediately skillful at using it | o | o | o | o | o |
| *Satisfaction* | | | | | | |
| 24 | I am satisfied with its working | o | o | o | o | o |
| 25 | I will suggest it to a friend | o | o | o | o | o |
| 26 | It is enjoyable to use it | o | o | o | o | o |
| 27 | It works as I wanted | o | o | o | o | o |
| 28 | It is wonderful | o | o | o | o | o |
| 29 | I feel the need to have it | o | o | o | o | o |
| 30 | It is pleasant to use | o | o | o | o | o |

*comprehensible*, *fast* and *efficient* represent the usability degree of the interface, while adjectives like *stimulanting*, *innovative* and *attractive* identify the capability of the prototype in terms of appealability. Our questionnaire is composed of 26 pairs of adjectives, chosen and validated by a group of experts in usability. For this questionnaire, the Likert scale provides for a score ranging from 1 to 7 (see Table 7). Half of the items include a negative adjective while the other half a positive one, so balancing the items submitted to the users. The order of negative and positive items is random.

**Table 7** The UEQ questionnaire

| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Uncomfortable | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Comfortable |
| 2 | Incomprehensible | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Comprehensible |
| 3 | Creative | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Not creative |
| 4 | Easy to understand | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Difficult to understand |
| 5 | Noticeable | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Poor |
| 6 | Boring | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Fascinating |
| 7 | Insignificant | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Interesting |
| 8 | Unpredictable | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Predictable |
| 9 | Fast | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Slow |
| 10 | Original | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Conventional |
| 11 | Obstructive | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Of support |
| 12 | Agreeable | ◦ | ◦ | ◦ ◦ | ◦ | ◦ | ◦ | ◦ | Disagreeable |
| 13 | Complicate | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Easy |
| 14 | Repellent | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Attractive |
| 15 | Usual | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Modern |
| 16 | Appreciated | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Unpleasant |
| 17 | Sure | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Unsure |
| 18 | Stimulanting | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Soporific |
| 19 | Satisfying | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Scant |
| 20 | Inefficient | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Efficient |
| 21 | Clear | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Messy |
| 22 | Not much practical | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Practical |
| 23 | Ordered | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Unordered |
| 24 | Attractive | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Not attractive |
| 25 | Friendly | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Hostile |
| 26 | Conservative | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | ◦ | Innovative |

## 6 Analysis of the responses collected by the different questionnaires

### 6.1 The USE questionnaire

Our USE questionnaire is able to provide a subjective evaluation of four different parameters. The responses given to the items that evaluate in an aggregate way one of the four parameters are analyzed in a cumulative modality. Specifically, as reported in Table 6, the items from 1 to 8 are collectively used to evaluate the usefulness, those from 9 to 19 are analyzed for the ease of use, the items from 20 to 23 evaluate the ease of learning and, finally, those from 24 to 30 take into account the user satisfaction.

**Table 8** The values of the mean *M* and standard deviation $\sigma$ of the 30 items related to the USE questionnaire

| Item | *M* | $\sigma$ | |
|---|---|---|---|
| 1 | 3.0 | 0.0 | Usefulness |
| 2 | 4.2 | 0.6 | |
| 3 | 4.5 | 0.7 | |
| 4 | 3.0 | 0.0 | |
| 5 | 4.0 | 0.8 | |
| 6 | 4.2 | 0.9 | |
| 7 | 4.1 | 0.6 | |
| 8 | 3.9 | 0.9 | |
| 9 | 3.0 | 0.0 | Ease of use |
| 10 | 4.3 | 0.7 | |
| 11 | 4.2 | 0.8 | |
| 12 | 3.7 | 1.0 | |
| 13 | 4.0 | 0.8 | |
| 14 | 3.0 | 0.0 | |
| 15 | 4.0 | 0.8 | |
| 16 | 3.0 | 0.0 | |
| 17 | 4.4 | 0.5 | |
| 18 | 3.7 | 0.8 | |
| 19 | 3.0 | 0.0 | |
| 20 | 4.2 | 0.8 | Ease of learning |
| 21 | 3.0 | 0.0 | |
| 22 | 4.4 | 0.7 | |
| 23 | 3.0 | 0.0 | |
| 24 | 4.2 | 0.7 | Satisfaction |
| 25 | 4.6 | 0.5 | |
| 26 | 4.3 | 0.7 | |
| 27 | 3.0 | 0.0 | |
| 28 | 3.0 | 0.0 | |
| 29 | 3.0 | 0.0 | |
| 30 | 3.0 | 0.0 | |

In Table 8 the values of the mean $M$ value and the standard deviation $\sigma$ of the 30 items, associated with usefulness, to the ease of use, ease of learning and satisfaction, have been reported to detect anomalous values in the evaluations. As it is possible to see, such values indicate that several questionnaire items collected from all the responses of the volunteers tend towards the middle score values of the Likert scale. These results are further discussed in Sect. 7.

### 6.2 The UEQ questionnaire

The UEQ questionnaire is expressly dedicated to an evaluation of the user experience, and scores are assigned to the items on the basis of the type of the question. In fact, if an item includes a negative adjective, its score is given by the value returned by the user minus 4. If, instead, an item includes a positive adjective its score is given by 4 minus the user value. The scores balanced in this way are used to obtain the final indicators for each item. When the scores are transformed on the basis of the above rules, the score + 3 represents the most positive score while − 3 the most negative one. We aggregated the results of the items associated with each of the six indicators. In Table 9 the grouping of items to the same indicator is evidenced through the use of different colors for each of these indicators. The detail of the relationship between the six indicators and the related items is the following:

– items 1, 12, 14, 16, 24 and 25 consider the attractiveness;

**Table 9** The values of the mean $M$ and standard deviation $\sigma$ of the 26 items

| Item | $M$ | $\sigma$ | Index | |
|---|---|---|---|---|
| 1 | 1.8 | 1.0 | Attractiveness | |
| 2 | 1.9 | 1.0 | Simplicity | |
| 3 | 1.1 | 1.9 | Innovation | |
| 4 | 1.2 | 1.9 | Simplicity | |
| 5 | 1.4 | 1.5 | Stimulation | |
| 6 | 1.6 | 1.1 | Stimulation | |
| 7 | 2.1 | 1.0 | Stimulation | |
| 8 | 0.0 | 1.2 | Reliability | |
| 9 | 0.9 | 1.3 | Efficiency | |
| 10 | 1.6 | 1.2 | Innovation | |
| 11 | 1.9 | 1.3 | Reliability | |
| 12 | 1.7 | 1.5 | Attractiveness | |
| 13 | 1.6 | 1.2 | Simplicity | |
| 14 | 1.7 | 1.2 | Attractiveness | |
| 15 | 1.8 | 1.3 | Innovation | |
| 16 | 2.0 | 1.0 | Attractiveness | |
| 17 | 1.3 | 1.5 | Reliability | |
| 18 | 1.7 | 1.3 | Stimulation | |
| 19 | 1.7 | 1.2 | Reliability | |
| 20 | 1.8 | 1.1 | Efficiency | |
| 21 | 1.8 | 1.2 | Simplicity | |
| 22 | 1.9 | 1.2 | Efficiency | |
| 23 | 1.5 | 1.3 | Efficiency | |
| 24 | 1.5 | 1.3 | Attractiveness | |
| 25 | 1.8 | 1.2 | Attractiveness | |
| 26 | 1.6 | 1.5 | Innovation | |

- items 2, 4, 13 and 21 account for the simplicity;
- items 9, 20, 22 and 23 consider the efficiency;
- items 8, 11, 17 and 19 relate to the reliability;
- items 5, 6, 7 and 18 take into account the stimulation;
- items 3, 10, 15 and 26 relate to the innovation.

In Table 9 the values of the mean *M* and standard deviation $\sigma$ of the 26 items have been analyzed to detect anomalous values in the evaluations.

The table shows that only question eight presents a mean score close to neutral while all the other scores are positive. These results obtained for each item support the conclusion that the App received a good evaluation, as discussed in Sect. 7.

The average values, restricted to a scale range from − 2 to + 2, for each analyzed aggregated indicator, are reported in Fig. 12. From this figure it is possible to see that the values for all the aggregated indicators are positive, so further demonstrating that our App meets users' approval. It can also be observed that the indicators that contribute to make our App interesting for the users are, mainly, its attractiveness and stimulation. Moreover, simplicity appears to be important in the positive evaluation. On the other hand, reliability seems to represent the least important aspect. However, a benchmark is needed to evaluate all the types of quality aspects of the user experience.

### 6.3 Benchmarking the user experience

The results shown in Fig. 12 do not have too much significance if considered on their own. In fact, they merely assert that user experience has been positive when measured on a numerical scale. However, considered alone, they cannot indicate how positive it might be with respect to that of other products. In order to make such an assessment, these results have to be compared against a benchmark (Schrepp et al. 2017). In fact, the question whether or not the user experience in relation to a new product is satisfactory should be addressed by means of a comparison of its

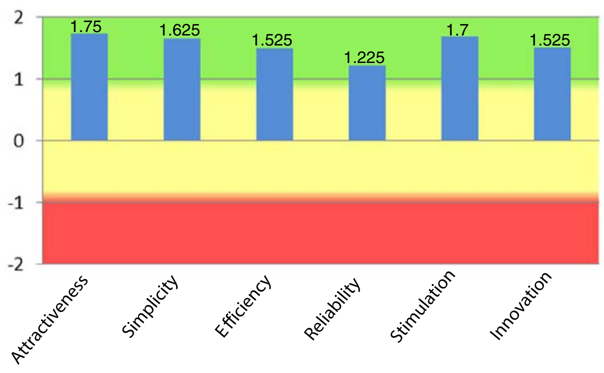



**Fig. 12** The average values outlined in a scale range limited from − 2 to + 2

results with those demonstrated by a large sample of other products, i.e., a benchmark data set. If the comparison reveals that the product under consideration has usability scores higher than those of the products contained in the benchmark, this is a good indication that the user experience of the product at hand can be considered as satisfactory. There is no need for all the products contained in the benchmark to be of the same type.

Several such benchmark data sets exist. We have considered here the recent benchmark proposed in Schrepp et al. (2017), that contains data coming from the evaluation of 246 products performed through the use of the UEQ questionnaire. The scale [− 3, + 3] for each of the six indicators considered in UEQ has been divided by the authors into five categories: excellent, good, above average, below average, and bad. The ranges for each such category vary from one parameter to another depending on the results contained in the benchmark data set for the considered indicator. The related ranges can be found in Table 1 of this seminal paper (Schrepp et al. 2017). By entering the six average values of the aggregated indicators reported in Fig. 12 into that table, we can observe that the benchmark results for GENnArí, shown in Table 10, are very good. By looking at these results, it can be concluded that our PETG is significantly above average, as it reaches at least the 'above average' score for all the indicators. Moreover, a very significant result of this analysis is that the feature indicators that most contribute to a positive user experience for GENnArí are its attractiveness, stimulation and innovation.

In Schrepp et al. (2017) it is stated that for new products the issues of simplicity, stimulation and innovation are the most important indicators when assessing user experience. Hence, they write that a sensible target when building a new application is that its evaluation in UEQ obtains the 'excellent' mark for these three indicators and at least an 'above average' for the remaining ones. Given the results we provide in Table 10, we can conclude that our GENnArí is close to excellence.

Although the results of the use of the UEQ questionnaire are quantitative, careful considerations of the six numerical values obtained and of the categories shown in Table 10 allows them to be used to make some educated guesses, on the one hand, about the features that most satisfy users, and, on the other hand, about those features for which improvements are more expected. The results reported in the table reveal that GENnArí is very satisfactory in terms of 'general attractiveness', and of 'hedonic quality', as shown by stimulation and innovation indicators. This latter suggests that GENnArí is innovative and creative, which means that its specific features as management of waiting times, personalized user speed, weather forecast, disabilities, and the use of augmented reality, smartwatch visualization, and 'Around you' functionality are welcome to users. On the other hand, the aspects related to

**Table 10** Results of user experience benchmarking

| Indicator | Attractiveness | Simplicity | Efficiency | Reliability | Stimulation | Innovation |
|---|---|---|---|---|---|---|
| Value | 1.75 | 1.625 | 1.525 | 1.225 | 1.7 | 1.525 |
| Category | Excellent | Good | Good | Above average | Excellent | Excellent |

'pragmatic quality', i.e., simplicity, efficiency and reliability could be improved. This is true especially with reference to reliability. This latter involves aspects as feeling in control of the interaction, predicting the system behavior, and feeling safe when working with GENnArí. Its relatively low value means that the user feels relatively unsure when using it, so that this issue should be improved, possibly by adding to the App a better description of the way it should be used. Moreover, simplicity involves the fact that getting familiar with all the new features possessed by GENnArí is not straightforward, whereas efficiency reveals that some effort is necessary to the users, that the interaction, although efficient and fast, could be improved, and that the product could react faster to user input. This latter means that the underlying optimization algorithm should be made faster, which will be an issue for our future work.

## 7 Discussion and lessons learned

The current version of our GENnArí App is still in its prototypal stage. Indeed, it still presents certain problems and is based over several simplifications and assumptions, as we discuss in this section. Similarly, its evaluation from the users has been carried out with the aim of receiving useful suggestions about how to improve it and eliminate any problems users could experience during its use. This means that until now we have been mostly motivated by the idea of creating a robust app and in debugging it in the field. Nonetheless, our interest also lies in evaluating the App prototype from an HCI point of view. Accordingly, meetings with HCI experts were also held to accept their suggestions so as to obtain an app that could be considered as user-friendly. In the following of this section we briefly discuss some of these issues that must be addressed in the near future.

*Personal scores.* The effectiveness of the personalized tours planned by the App heavily depends on the accuracy of the data supplied to the optimizer. The personal scores and the information related to the events that can influence the visit (for example opening, closing, queueing and visiting times, and weather forecasting) are crucial. The evaluation of accurate values for the personal score associated with each *POI* relies on the goodness of the profiling phase perfromed. In the paper the availability of these values, as detailed in Sect. 3.2.3, is assumed. We are aware that more reliable values can be automatically obtained by exploiting content, collaborative, demographic and hybrid filtering techniques (Anacleto et al. 2014; Tewaria and Barman 2018; Chen and Tsai 2019). Besides, it would be possible to consider factors such as popularity, distance and itinerary travel time (Kotiloglu et al. 2017), the combination of geo-tagged web photos and collaborative filtering (Jiang et al. 2013) and the fusion of geographical and user social network information (Gao et al. 2018) to model the *POI* score. Therefore, the lack of an automatic methodology to rank personalized tourism attractions is a practical but not a conceptual limitation of the current PETG version.

*Waiting times* Another technical limitation is the information related to the real average waiting times that we assume to have at our disposal during the organization of the personal tour. With regards to their estimation, several methods have been proposed to derive reliable values. One example is shown in Meys and Groen (2014), in which tourists are provided with real-time information on both the queue length and the related estimated waiting time at the Van Gogh museum in Amsterdam. The waiting time estimation has been performed thanks to the use of a sensor-based system containing four sensor nodes, a base station, and an antenna. This method also allows the creation of tables with average waiting times for different days and hours at each *POI*. Another noticeable estimation system is provided in Shu et al. (2016), where a non-standard autoregressive model is proposed and used in conjunction with Wi-FI positioning. This system learns through the use of both Wi-Fi estimation results from the previous day and the current queuing time so as to forecast the queuing time in the forthcoming time period. A several-days test was carried out at Beijing Capital International Airport with excellent results. In (Liao and Zheng 2018) the authors suggest a method to evaluate waiting times, due to the fact that the queue and the site capacity are time-dependent stochastic variables that follow diverse types of distribution functions in different time slots. Such functions can be fitted in accordance with a large amount of historical data. On our part, to estimate the average waiting times, it is sufficient to install detection systems at each *POI* to retrieve real or quasi-real time values in a specific hour or time slot. Therefore, we are working with the institutions that manage the different *POI*s to obtain an updated estimation for those times. This could be realized by means of numerical information, such as the number of sold tickets, and/or through sensors or cameras placed at the entrances. Naturally this data can be stored in a database and recovered by our App through network queries.

We feel that the approach developed by Meys and Groen could be easily implemented by the Naples *POI* managers, and could prove very useful for us in terms of hourly waiting times.

*Visiting times* Similarly, several methods exist in the literature to estimate the visiting times. In Cotfas et al. (2011) the authors propose a formula for the evaluation of these times. Specifically, for each user this formula modifies the average visiting time for a given *POI* $p_i$ by multiplying this by a factor. This latter factor takes into account the differences in the amounts of time spent by this user in visiting other POIs that belong to the same category as $p_i$ with respect to the average visiting times for those POIs. This means that the longer is the time spent for those *POI*s similar to $p_i$, the longer will be the visiting time predicted for this specific *POI*.

In Brilhante et al. (2015) a method to estimate visiting times is outlined. In particular, the method utilizes user-generated contents on a personal photo sharing social network to retrieve the metadata associated with the photos taken in a given area for a specific *POI*. The authors consider the first and the last photos taken by the same user as the start and end times of the user's visit to the *POI*. The visiting time is then assessed by evaluating for each *POI* the average of these times. It is evident that this method is based on strong and random assumptions to too great a degree to be considered reliable.

In Migliorini et al. (2018), the authors use an offline analysis to analyze visits to *POI*s. Specifically, people who buy the VeronaCard tourist card are considered. These tourists have to timestamp their card at the gates when entering and exiting *POI*s, and therefore the authors can reconstruct the set of visited sites. The popular tours are stored in a database queried by a recommendation engine. From the data, it is also possible to derive a set of characterizing measures, such as the average number of visitors inside each *POI* at different times and the average visit duration given the number of visitors.

In Liao and Zheng (2018), Zheng and Liao (2019) the authors consider as the *POI* visiting time the average time spent by previous tourists at the site. To collect data on these times, the answers from questionnaires distributed to tourists are combined with the information gathered from the site staff members. Tourists who were leaving the site were randomly approached and asked about any *POI*s visited and the approximate amount of time they had spent at each. As suggested by the staff members, any values that are significantly greater or smaller than those of most tourists can be regarded as anomalous and should be removed. The average visit duration for each POI is evaluated as the average of the remaining values.

From the approaches presented above, we can conclude that the problem of estimating the visiting time is also solvable and thus simply represents a technical limitation of our current guide prototype. We plan to follow the approach devised by Cotfas et al. (2011), as it is the simplest one, requiring neither hardware nor software nor questionnaires.

*Around you* During the App evaluation, one of the problems experienced regards the notifications enabled by the users for the 'Around you' service. We noticed that, given the high concentration of *POI*s, in Naples city center, many notifications were generated during the tours. This frequency became irritating fro some users, so that many preferred to disable it, while others decided to keep it enabled with a very limited distance threshold, say 10 m. This latter is lower than the default threshold distance of 50 m set in the App.

*Evaluation analysis* From the analysis of the responses to the questionnaires it is possible to learn some lessons that will assist us in improving the functionalities and the interaction modalities of the App. First of all, many of the USE questionnaire items collected during the evaluation present middle point scores. This means that the associated questions are not useful for an effective App evaluation. For example, this limitation applies in particular in the case of item 1 ('it helps to enhance the benefits of the cultural heritage'). Rightly the users stated that GENnArí does not support the enjoyment of the cultural heritage because the information provided is minimal, and the user, once inside the *POI*, is not guided. Achieving such an objective would require a further effort in terms of providing enriched cultural heritage information. As a further example, four out of the seven items of the satisfaction indicator are poorly informative in the evaluation. Specifically, the reply to the item 27 ('it works as I wanted') should have been easily predictable: the user is not an expert and therefore did not have much idea as to what to expect. The item 28 ('it is wonderful') seems self-promoting. On the contrary, many responses concerning the ease of use and the ability of GENnArí to save time in planning a tour and in stimulating curiosity about the cultural heritage, although not directly enhancing its benefits, are above the middle score, so indicating the validity of the App.

## 8 Conclusions and future works

In this paper an interactive mobile electronic guide application, namely 'GENnArí' App, is presented: its interactive functionalities and facilities are illustrated, and its capability in planning personalized multiple-day tourist tours by taking into account different contrasting objectives is described. The tour optimizer application automatically plans the itinerary by selecting the sights of potential interest based on user preferences and constraints such as the available visit time on a daily basis, opening days and operating hours, average visiting times, accessibility of the places of interests, and weather forecasting. Such an optimizer relies on an evolutionary algorithm that is able to provide near-optimal user-adapted tourist routes in reasonable times.

Furthermore, an experimental analysis has been performed to validate the usability and perceived utility of the advanced application prototype with particular reference to the interactive functionalities and facilities offered together with user satisfaction. The analysis has been carried out by means of two different questionnaires, namely the USE and UEQ, completed by volunteers during some public demonstrations of the prototype. The results of this on-field user evaluation study are shown in this paper, and evidence that our guide is evaluated positively by its users in terms of usefulness, ease of use, ease of learning and satisfaction. Moreover, a benchmarking for the user experience has been performed to assess the quality of the App. This benchmarking allows an identification of the features which are most meaningful in relation to the user experience.

It is worth noting that GENnArí can be very easily modified and adapted to propose walking itineraries in the historical centers of any other city. All that is necessary is to equip the guide with the databases of the local *POI*s and the weather forecasting information.

Our future works will involve the implementation of a more efficient profiling framework capable of deriving personalized information by exploiting collaborative filtering, or content- and knowledge-based paradigms, or hybrid and demographic filtering applications as discussed in Sect. 7. This will also include the design of an automatic tool to acquire the personal score for the *POI*s. A further issue to consider is the adoption of a methodology to evaluate reliable waiting and visiting times at the sites.

We will try to limit the information directly asked to the user by implementing methods for the automatic detection of user features such as the average walking speed. Moreover, we will add the tourist's available budget as a further constraint to take into account. Finally, we plan to improve the functionalities of the App by exploiting the results of an in-depth examination of the usability of augmented reality and smartwatch interfaces.

**Acknowledgements** This work has been supported by the project "Organization of Cultural Heritage for Smart Tourism and Real-Time Accessibility (ORCHESTRA)" (PON04a2_D) approved and financed within the 2012 "Smart Cities and Communities" call of the Italian Ministry for University and Research.

## References

Anacleto R, Figueiredo L, Almeidaa A, Novais P (2014) Mobile application to provide personalized sightseeing tours. J Netw Comput Appl 41:56–64
Ardissono L, Kuflik T, Petrelli D (2012) Personalization in cultural heritage: the road travelled and the one ahead. User Model User-adapt Interact 22(1–2):73–99
Betram D (2009) Likert scales. The Faculty of Mathematics University of Belgrad, Croatia, Technical report
Borràs J, Moreno A, Valls A (2014) Intelligent tourism recommender systems: a survey. Expert Syst Appl 41(16):7370–7389
Brilhante IR, Macedo JA, Nardini FM, Perego R, Renso C (2015) On planning sightseeing tours with T ripB uilder. Inf Process Manag 51(2):1–15
Brown B, Chalmers M (2003) Tourism and mobile technology. In: Proceedings of the 8th European conference on computer supported cooperative work (ECSCW), Springer, pp 335–354
Chen CC, Tsai JL (2019) Determinants of behavioral intention to use the personalized location-based mobile tourism application: an empirical study by integrating TAM with ISSM. Future Gener Comput Syst 96:628–638
Coello ACC, Van Veldhuizen AD, Lamont GB (2007) Evolutionary algorithms for solving multi-objective problems, vol 2. Genetic and evolutionary computation series. Springer, New York
Cotfas LA (2011) Collaborative itinerary recommender systems. Econ Inform J 11(1):191–200
Cotfas LA, Diosteanu A, Dumitrescu SD, Smeureanu A (2011) Semantic search itinerary recommender systems. Int J Comput 5(3):370–377
De Falco I, Scafuri U, Tarantino E (2015) A multiobjective evolutionary algorithm for personalized tours in street networks. Lecture notes computer science. Springer, New York, pp 115–127
De Falco I, Scafuri U, Tarantino E (2016) Optimizing personalized touristic itineraries by a multiobjective evolutionary algorithm. Int J Inf Technol Decis Mak 15(6):1269–1312
Deb K (2001) Multi-objective optimization using evolutionary algorithms. Wiley, Chichester
Diosteanu A, Cotfas LA, Smeureanu A, Dumitrescu SD (2011) Natural language processing applied in itinerary recommender systems. In: Proceedings of the 10th international conference on applied computer and applied computational science, WSEAS Press, pp 260–265
Dix A, Finlay J, Abowd GD, Beale R (2004) Human–computer Interaction, vol 3. Pearson-Prentice Hall, New Jersey
Expósito A, Mancini S, Brito J, Moreno JA (2019) A fuzzy GRASP for the tourist trip design with clustered POIs. Expert Syst Appl 127(1):210–227
Feo TA, Resende MGC (1989) A probabilistic heuristic for a computationally difficult set covering problem. Oper Res Lett 8:67–71
Fogli A, Sansonetti G (2019) Exploiting semantics for context-aware itinerary recommendation. Pers Ubiquit Comput. https://doi.org/10.1007/s00779-018-01189-7
Gambardella L, Montemanni R, Weyland D (2012) Coupling ant colony systems with strong local searches. Eur J Oper Res 220(3):831–843
Gao R, Li J, Li X, Song C, Zhou Y (2018) A personalized point-of-interest recommendation model via fusion of geo-social information. Neurocomputing 273:159–170
Garcia A, Linaza MT, Arbelaiz O, Vansteenwegen P (2009) Intelligent routing system for a personalized electronic tourist guide. In: Höpken W, Gretzel U, Law R (eds) Information and communication technologies in tourism 2009. Springer, New York, pp 185–197
Garcia A, Vansteenwegen P, Arbelaitz O, Souffriau W, Linaza MT (2013) Integrating public transportation in personalised electronic tourist guides. Comput Oper Res 40(3):758–774
Garcia I, Sebastia L, Onaindia E (2011) On the design of individual and group recommender systems for tourism. Expert Syst Appl 38:7683–7692
Gavalas D, Kenteris M, Konstantopoulos C, Pantziou G (2012) Web application for recommending personalised mobile tourist routes. IET Softw 6(4):313–322
Gavalas D, Konstantopoulos C, Mastakas K, Pantziou G (2014a) Mobile recommender systems in tourism. J Netw Comput Appl 39:319–333
Gavalas D, Konstantopoulos C, Mastakas K, Pantziou G (2014b) A survey on algorithmic approaches for solving tourist trip design problems. J Heuristics 20(3):291–328
Gavalas D, Kasapakis V, Konstantopoulos C, Pantziou G, Vathis N, Zaroliagis C (2015a) The eCOMPASS multimodal tourist tour planner. Expert Syst Appl 42(21):7303–7316

Gavalas D, Konstantopoulos C, Mastakas K, Pantziou G, Vathis N (2015b) Heuristics for the time dependent team orienteering problem: application to tourist route planning. Comput Oper Res 62:36–50
Gavalas D, Kasapakis V, Konstantopoulos C, Pantziou G, Vathis N (2017) Scenic route planning for tourists. Pers Ubiquit Comput 21(1):137–155
Gunawan A, Lau HC, Vansteenwegen P (2016) Orienteering problem: a survey of recent variants, solution approaches and applications. Eur J Oper Res 255(2):315–332
Hu Q, Lim A (2014) An iterative three-component heuristic for the team orienteering problem with time windows. Eur J Oper Res 232:276–286
Huang Y, Bian L (2009) A bayesian network and analytic hierarchy process based personalized recommendations for tourist attractions over the internet. Expert Syst Appl 36(1):933–943
Hyde KK, Lawson R (2003) The nature of independent travel. J Travel Res 42(1):13–23
Jiang K, Yin H, Wang P, Yu N (2013) Learning from contextual information of geo-tagged web photos to rank personalized tourism attractions. Neurocomputing 119(7):17–25
Kenteris M, Gavalas D, Pantziou G, Konstantopoulos C (2010) Near-optimal personalized daily itineraries for a mobile tourist guide. In: Proceedings of the symposium on computers and communications (ISCC), IEEE, pp 862–864
Kenteris M, Gavalas D, Economou D (2011) Electronic mobile guides: a survey. Pers Ubiquit Comput 15(1):97–111
Kotiloglu S, Lappas T, Pelechrinis K, Repoussis P (2017) Personalized multi-period tour recommendations. Tour Manag 62:76–88
Kou G, Lin C (2014) A cosine maximization method for the priority vector derivation in AHP. Eur J Oper Res 235(1):225–232
Kou G, Lu Y, Peng Y, Shi Y (2012) Evaluation of classification algorithms using MCDM and rank correlation. Int Technol Decis Mak 11(1):197–225
Kou G, Peng Y, Wang G (2014) Evaluation of clustering algorithms for financial risk analysis using MCDM methods. Inf Sci 275:1–12
Labadie N, Mansini R, Melechovský J, Calvo RW (2012) The team orienteering problem with time windows: an LP-based granular variable neighborhood search. Eur J Oper Res 220:15–27
Laugwitz B, Held T, Schrepp M (2008) Construction and evaluation of a user experience questionnaire. Springer, Berlin
Liao Z, Zheng W (2018) Using a heuristic algorithm to design a personalized day tour route in a time-dependent stochastic environment. Tour Manag 68:284–300
Lin SW, Yu VF (2012) A simulated annealing heuristic for the team orienteering problem with time windows. Eur J Oper Res 217:94–107
Lourenco HR, Martin O, Stuetzle T (2002) Iterated local search. In: Glover F, Kochenberger GA (eds) Handbook of metaheuristics. Kluwer Academic Publishers, Norwell, pp 321–353
Lund AM (2001) Measuring usability with the use questionnaire. Usability Interface 8(2):3–6
Meys W, Groen M (2014) Quo vadis?: Persuasive computing using real time queue information. In: ACM (ed) Proceedings of the first international conference on IoT in urban space, pp 102–104
Migliorini S, Carra D, Belussi A (2018) Adaptive trip recommendation system: balancing travelers among pois with MapuppercaseReduce. In: Proceedings of the IEEE international congress on big data, IEEE, pp 255–259
Montemanni R, Weyland D, Gambardella L (2009) Ant colony system for the team orienteering problem with time windows. Found Comput Decis Sci 34:287–306
Montemanni R, Weyland D, Gambardella L (2011) An enhanced ant colony system for the team orienteering problem with time windows. In: Proceedings of the international symposium on computer science and society, IEEE, pp 381–384
Muccini H, Rossi F, Traini L (2017) A smart city run-time planner for multi-site congestion management. In: Proceedings of the international conference on smart systems and technologies (SST), IEEE, pp 175–179
Nielsen J (1993) Usability engineering. Academic Press, Cambridge
Preece J, Rogers Y, Sharp H (2002) Interaction design, beyond human-computer interaction. Wiley, New York
Righini G, Salani M (2009) Decremental state space relaxation strategies and initialization heuristics for solving the orienteering problem with time windows with dynamic programming. Comput Oper Res 36(4):1191–1203

Rodríguez B, Molina J, Pérez F, Caballero R (2012) Interactive design of personalised tourism routes. Tour Manag 33:926–940
Schrepp M, Hinderks A, Thomaschewski J (2017) Construction of a benchmark for the user experience questionnaire (UEQ). Int J Interact Multimed Artif Intell 4(4):40–44
Shu H, Song C, Pei T, Xu L, Ou Y, Zhang L, Li T (2016) Queuing time prediction using WiFi positioning data in an indoor scenario. Sensors 16(11):E1958
Souffriau W, Vansteenwegen P, Vertommen J, Berghe GV, Van Oudheusden D (2008) A personalised tour trip design algorithm for mobile tourist guides. Appl Artif Intell 22(10):964–985
Souffriau W, Maervoet J, Vansteenwegen P, Vanden Berghe G, Van Oudheusden D (2009) A mobile tourist decision support system for small footprint devices. In: Lecture notes computer science, bio-inspired systems: computational and ambient intelligence, LNCS 5517, Springer, New York, pp 1248–1255
Souffriau W, Vansteenwegen P, Vanden Berghe G, Van Oudheusden D (2013) The multiconstraint team orienteering problem with multiple time windows. Transport Sci 47(1):53–63
Sylejmani K, Kosova P, Dorn J, Musliu N (2012) A tabu search approach for multi constrained team orienteering problem and its application in touristic trip planning. In: Proceedings of the 12th international conference on hybrid intelligent systems, IEEE, pp 300–305
Sylejmani K, Dorn J, Musliu N (2017) Planning the trip itinerary for tourist groups. Inf Technol Tour 17(3):275–314
Tewaria AS, Barman AG (2018) Sequencing of items in personalized recommendations using multiple recommendation techniques. Expert Syst Appl 97:70–82
Tricoire F, Romauch M, Doerner K, Hartl R (2010) Heuristics for the multi-period orienteering problem with multiple time windows. Comput Oper Res 37:351–367
Umanets A, Ferreira A, Leite N (2014) GuideMe—a tourist guide with a recommender system and social interaction. Proc Technol 17:407–414
Vansteenwegen P, Souffriau W, Van Oudheusden D (2009a) A detailed analysis of two metaheuristics for the team orienteering problem, engineering stochastic local search algorithms. Lecture notes computer science, vol 5752. Springer, New York, pp 110–114
Vansteenwegen P, Souffriau W, Vanden Berghe G, Van Oudheusden D (2009b) Iterated local search for the team orienteering problem with time windows. Comput Oper Res 36(12):3281–3290
Vansteenwegen P, Souffriau W, Van Oudheusden D (2011a) The orienteering problem: a survey. Eur J Oper Res 209(1):1–10
Vansteenwegen P, Souffriau W, Vanden Berghe G, Van Oudheusden D (2011b) The city trip planner: an expert system for tourists. Expert Syst Appl 38(6):6540–6546
Verbeeck C, Sörensen K, Aghezzaf EH, Vansteenwegen P (2014) A fast solution method for the time-dependent orienteering problem. Eur J Oper Res 236(2):419–432
Wik (2017) Wikitude developer SDK. http://www.wikitude.com/developer/documentation/android. Accessed 15 June 2019
Yon H, Zheng Y, Xie X, Woo W (2012) Social itinerary recommendation from user-generated digital trails. Pers Ubiquit Comput 16(5):469–484
Zheng W, Liao Z (2019) Using a heuristic approach to design personalized tour routes for heterogeneous tourist groups. Tour Manag 72:313–325