Active Aging by Continuous Learning: A Training Environment for Cultural Visits

**Amedeo Cesta, Gabriella Cortellessa, Riccardo De Benedictis
and Francesca Fracasso**

Abstract The "Città Educante" project aims at radically rethinking the learning environments through the application of the most advanced ICT technology. Among the different aspects of the project, the *LECTurE* module aims, by exploiting artificial intelligence techniques, at proposing contextualized lessons, as well as interaction requests, to the involved users. In particular the lesson's content, is specifically tailored for the older adults and personalized by taking into account users' psycho-physiological aspects as well as geo-localization information and temporal constraints. In this paper, after a generic introduction to the "Città Educante" project, the *LECTurE* module is presented and instantiated in two relevant use cases: the *on-site training*, in which the system is used as a support to the classical teaching methodologies within a classroom, and the *distributed training*, in which the technology aims at moving and animating the teaching experience, also, "outside the classroom", with additional stimuli for the users during a practical experience (e.g., a real visit in a museum to complement a theoretical art history lesson). The paper describes, then, the choices made for the realization of a first prototype and its embodiment into a concrete scenario which, by implementing a "treasure hunt" like game, aims at fostering older adults' physical and cognitive activity.

Keywords Intelligent tutoring system · Active assisted living · Active ageing
Planning and scheduling

A. Cesta · G. Cortellessa · R. De Benedictis (✉) · F. Fracasso
CNR - Italian National Research Council, ISTC, Rome, Italy
e-mail: riccardo.debenedictis@istc.cnr.it

A. Cesta
e-mail: amedeo.cesta@istc.cnr.it

G. Cortellessa
e-mail: gabriella.cortellessa@istc.cnr.it

F. Fracasso
e-mail: francesca.fracasso@istc.cnr.it

1 Introduction

By exploiting the most advanced ICT technology, the "Città Educante" project[1] (the name, in Italian, means "city that educates") aims at completely rethinking the concept of *learning environment* while reshaping it according to a smart city perspective. The project, specifically, proposes new educational approaches, while enriching and innovating methods and tools, by providing a learning environment which acts in time (e.g., lifelong) and space (e.g., at school, in an outdoor environment, during leisure, etc.) hence overcoming the classical systems and the traditional "lessons".

The theme of *learning* is, through the project, framed in relation to the response to social challenges linked to the renewal of the educational system, to be achieved by means of the implementation of new learning/teaching models and/or the optimization of the existing ones on the various areas of life and knowledge, as well as new systems/evaluation processes, in which the technology (platforms and web) becomes an enabling factor.

Specifically, the authors' goal in the project has been the one of thinking an incarnation of the "Città Educante" for the continuous education of older people. In particular, considering the more recent experiences in the interaction of elderlies with complex machines [4], keeping in mind previous experience in training, specifically tailored for crisis managers [1], we are currently pursuing the goal of building an Intelligent Tutoring System (ITS) [9, 10], called *LECTurE* (for Learning Environment "Città Educante"), aimed at improving the active aging and the participation in the social life of the elders living at home, in the community, and at work.

This paper represents an introduction to authors' effort in designing and building the *LECTurE* learning subsystem. After introducing, in Sect. 2, the main ideas underlying the learning subsystem, Sects. 3.1 and 3.2 describe a proposal for the modeling of students and lessons. A preliminary system architecture is proposed in Sect. 4. Section 5 describes the application of the *LECTurE* system to a concrete example. Finally, some conclusions and a discussion about future works close the paper.

2 Using AI to Personalize Lessons for Older Users

While pursueing the objective of the project, which aims at reformulating the learning environments through the creation of platforms, services and ICT applications, we got in touch with several volunteering organizations addressing, specifically, elderlies' needs. Among the different organizations, in particular, Televita[2] is a volunteering association whose main objective is to maintain the elderlies active and motivated, leveraging upon individual aptitudes and/or competencies [3]. Televita's volunteers, in fact, are, themselves, elders who want to keep active by offering their abilities

---

[1] http://www.cittaeducante.it.
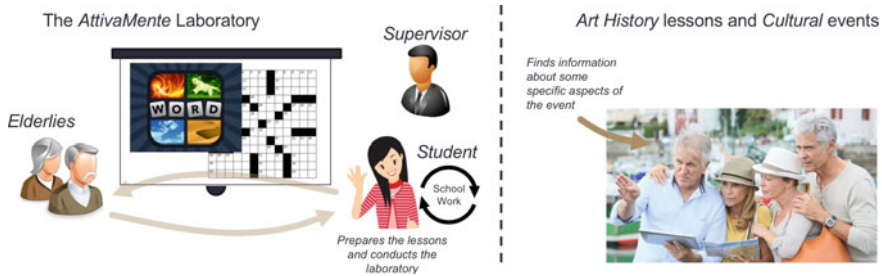
[2] http://www.televita.org.

Fig. 1 The *LECTurE* general idea

and competences to the organization. Although Televita's main activity consists in providing tele-assistance services (tele-friendship) and a 24/7 active helpline devoted to lonely elders who need support, it also manages several laboratories that involve elders both as attendees and "teachers". Examples include a computer lab, a tailoring lab, a cooking lab and an Italian language teaching for foreign people. Furthermore, the association organizes cultural events as concerts, museum visiting, theater, etc.

Among the offered services, we focused in particular on two specific activities that were both in line with the "Città Educante" concept and that are outlined in Fig. 1:

- The *AttivaMente laboratory*: aims at keeping elders mentally active, so as to limit the cognitive decline associated with the advancement of age, by proposing them cognitive stimuli. Such stimuli, mostly consisting in general culture quiz and/or crosswords, are proposed to the elders in a context similar to a school lesson. Specifically, by relying on some previous knowledge of the involved persons, as well as on their interactions during the lesson, a teacher, a volunteer himself, yet with more experience than others, controls the course's progress and slightly adapt it to the specific context's needs. Since the stimuli are predetermined, however, such adaptations tend to be limited to the possibilities of the case. The use of artificial intelligence techniques, in this case, could support the personalized delivery of stimuli by taking into account previous knowledge of the participants as well as their interactions with the AI system increasing, compared to the classical case, the personalization capabilities.

- The *Art History lessons and cultural events*: analogously to the AttivaMente case, before attending to cultural events, some of the participants, according to their abilities and competencies, are asked to find information about some specific aspects of the event (e.g., a particular work of art within a museum, relevant historical happenings related to a visited site, etc.). Such information are then shared, in a "sort of" lesson with other participants in a classroom context and also outside, during the event, enriching the overall knowledge of the group while encouraging the interaction among the members. Similarly to the AttivaMente case, such information may result to be limited and/or not customized to the specific members of the event. AI techniques, in this case, might offer the opportunity to

enrich the cultural events experience by providing further personalized stimuli to the event participants, as well as interaction requests to test the level of engagement and actively stimulate them.

Combining the above activities represents an opportunity for the elders to keep themselves active while learning in time and space. Additionally, the use of AI can support the development of more effective and engaging learning experience.

2.1 The LECTurE *Learning Subsystem*

The idea of developing the *LECTurE* system has arisen as a consequence of a field experience. Specifically, by taking inspiration from a previous work [2], in which students were trained for managing crisis, the approach used within *LECTurE* is based on the idea of dynamically composing lessons through the use of a technology related to a subfield of artificial intelligence called *automated planning* [6]. In particular, starting from a static representation containing an high-level lesson track, initially stored in a database, the lesson is *planned* and dynamically adapted and personalized to the involved users. The idea of using the technology related to automated planning comes from the need to create a sufficiently extensive didactic experience to reproduce a large number of different situations which are, at the same time, characterized by a high variability of stimuli, aimed at increasing the involvement level of users. Automated planning, indeed, favors the generation of different lessons that would be too complicated to obtain with a simple pre-compilation of stories. The timeline-based approach to automated planning [8], in particular, represents the unifying element of the various modules by ensuring the dynamic adaptability of plans by promoting experiential learning.

From a high-level point of view, the main modules of the system are described in Fig. 2. In particular, it is possible to distinguish between two kinds of **involved users**: the *students*, i.e., a group of people, potentially, of any age, interested in using the learning services offered by the *LECTurE*, and the *teachers*, i.e., users with special privileges who have the opportunity to observe students, monitoring the progress of the lessons and of the overall learning environment. The above users interact with the *LECTurE* system which is composed of three functional blocks, intended as architectural subsystems, implementing the corresponding high-level functionalities: (i) the *user modeling*, whose goal is to create and maintain a user model and provide guidance for improving the learning process; (ii) the *lesson modeling*, whose role consists in combining the information from the previous subsystem and to create the customized lesson as well as to control its evolution; (iii) the *lesson presentation*, whose purpose is to effectively represent the lesson.

It is worth highlighting that the proposed system provides users, whether students or teachers, the opportunity to change the learning environment's evolution in real time by interpreting their decisions. In fact, the architecture is based on a sense-plan-act paradigm implementing, in a continuous loop, the three primitives (a) *sense*, in
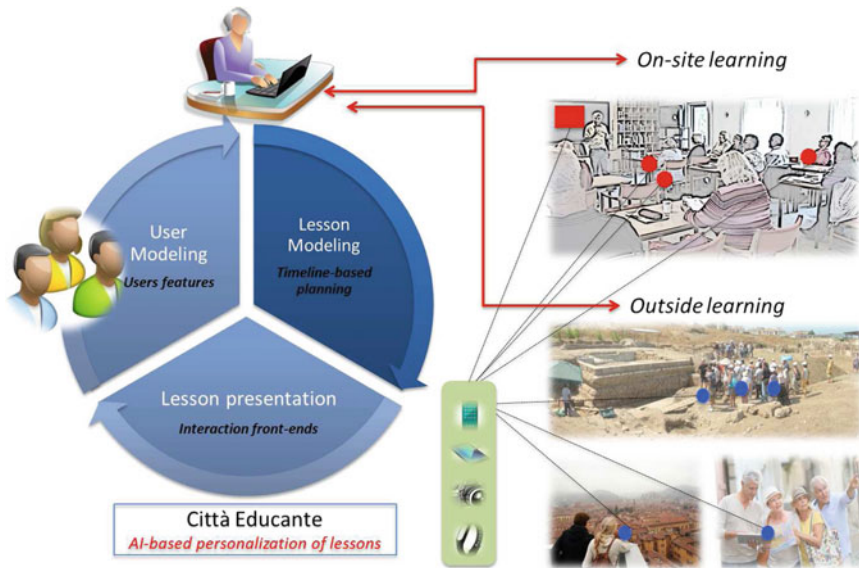
Fig. 2 The *LECTurE* general idea

which information is collected from sensors, (b) *plan* where a world model is created using the information available to plan the next move to do, and (c) *act*, in which the move, chosen by the planning process, is actually executed. Furthermore, by mimicking the different cases implemented by the Televita organization, the *LECTurE* system provides training through different modalities depending on the chosen use case. In particular, it is possible to distinguish between two different modalities: (a) *on-site learning*, closer to the classical teaching, in which technology is used as a support to the teaching in a classroom, with the aim to create richer lessons, and (b) *distributed learning*, in which technology aims to support lessons outside the classroom during a practical experience. The following sections describe, more in detail, the above modalities.

2.2 On-Site Training

In the on-site training modality, the system can be used by a group of students, at the same time, previously contacted by the teacher. This mode represents an extension to the classical learning method in which a teacher teaches to a group of students. In this case, however, compared to the classical approach, the teaching is enhanced by the introduction, within the lesson, of the *LECTurE* technology.

Specifically, each lesson is instantiated by the teacher by defining the specific *learning objectives* [7]. The system processes the lesson and presents the information to the students through the available tools. Students interact directly with the

system, providing their answers to certain circumstances proposed by the system, and transmitting data from sensors available on the adopted devices (e.g., physiological parameters) enriching the users' models.

Depending on the sensors' inputs and on the students responses, both the teacher and the system can autonomously decide, based on the observations, whether and how to modify the current lesson. Finally, at the end of the lesson, the system can generate several reports, for example, one for each student, which can be used for debriefing purposes.

2.3   Distributed Training

In the distributed training case, the system exploits different types of web technologies. This means that the lesson does not happen in a single physical room and is distributed among the students who are remotely connected to the system. The lesson can still be instantiated by the teacher defining the specific learning objectives but may have variable, potentially infinite, duration.

This kind of approach, compared to the previous one, is more innovative. Students interact directly with the system while wandering, by their own, within the city, providing their answers to certain circumstances proposed by the system as well as constantly transmitting data from the sensors available on the chosen devices (e.g., geographic location, physiological parameters, etc.). Sensor data, in particular, enriches the users' models which, in turn, adapt the lesson to the students resulting in a highly personalized learning experience.

Again, based on the inputs from the sensors and on the students' feedback, both the teacher and the system can autonomously decide, based on the observations, whether and how to modify the lesson in progress. Similarly to the on-site case, the *LECTurE* system can generate, during the lesson, several reports, for example, one for each student, which can be used both for debriefing purposes and for a further tuning of the lesson.

3   The *LECTurE* System

As already mentioned, the *LECTurE* users interact with the system which is composed of different functional blocks. In particular, the user modeling module aims at creating and maintaining a model of the users which is used as a starting point for the personalization of the lessons. The following sections describe, in more detail, how students and lessons are modeled and how the two models interact so as to guarantee the customization of the learning experience.

3.1 Modeling the Students

By pursuing the overall goal of enhancing the learning experience, it is necessary to keep a user model up-to-date in order to consider how their emotional, psychological, physiological and geographical parameters can influence the learning process. Specifically, the student modeling module has three main objectives:

- Select, model and monitor relevant human factors, as well as psychological, physiological, or other user-related variables;
- Develop a model that can represent the user's profile;
- Provide a high level guidance for customizing learning objectives.

The set of considered relevant factors include, among the other things, personality traits, past experience, the perceived effectiveness in performing complex tasks, the perceived fatigue, the level of engagement and the current performance assessment. The use of Blue-tooth bracelets (e.g. the Empatica E4), for example, allows the extraction of physiological values such as temperature, skin conductance, heart rate and heart variability. Additionally, it is possible to leverage on geo-localization services to get a good estimate of the users position in time. The initial evaluation of these variables, used as a baseline to initialize the didactic experience and as a reference point for subsequent measurements, can be done through the use of standardized psychological questionnaires or physiological measurements performed off-line before the lesson. It is worth to notice, however, that the profile of a student can also be updated exploiting the interactions of the users with the system asking them, for example, to answer to sporadic questions. As an example, the users' *engagement* is measured through a five levels Likert-type scale which is administered to the users at regular intervals.

Among the dynamic parameters, particular emphasis is given to the students' *performance* which is monitored and observed by recording input data such as the different actions taken by the users. Specifically, this information is processed and interpreted in order to plan the subsequent actions of the lesson as well as to support the debriefing phase. By processing the users' profiles, the system generates a user model that is constantly updated to perceive and represent significant changes in the emotional state (note that parameters can generally change over time). In addition, students' performance is analyzed and processed in order to gather further usable information to customize even more the lesson. The teacher can access this information in order to supervise and control the customization. For this purpose, this component can provide guidance on how to customize the lesson. Personalization of a training course can therefore be done automatically, but it can also be suggested to a teacher who independently decides whether to adapt the training course (i.e., according to a mixed-initiative style).

Since the different users are characterized by properties that are closely related to their role, their work or, more generally, their psychophysiological state during the lessons, each user that interacts with the system feeds the *LECTurE* system with personal data. The system, in turn, builds a user model that is used to synthesize

custom lessons responding to the specific status of each student. The output of this process is passed to a second module that, on the basis of these indications and on the particular didactic path chosen, synthesizes a sequence of appropriate stimuli for the group (shared information between the different students in a lesson) and for the individuals (tailored information for a specific student).

3.2 Modeling the Lessons

Lesson modeling is the key feature of the *LECTurE* system since it creates and manages the network of events that guides the entire learning session. Nodes on this network are intended to represent temporally annotated stimuli (e.g., videos, text messages, questions, etc.) to be sent, at appropriate time, to their associated users while edges represent causal and temporal relations among such stimuli. Additionally, the network allows to the teacher to maintain a high-level vision of the lesson while providing sufficient granularity of the information sent to the students.

It is worth noting that although the above network is initialized in order to represent an abstract blueprint of a lesson, it is afterwards customized and dynamically adapted to the profile of the involved user. Specifically, adaptations to the network are made thanks to the application of a set of pre-compiled first-order-like *rules* which define how to "react" to the users' profile, to their updates and to their actions (e.g., answering to a question). Such rules, in particular, are intended to create the "conditions", in terms of events and their relations within the network, for other events to be present. An example of rule can be "in order to stimulate the cognitive activity of the group, either propose a simple crosswords and the group's performance is low, or propose a complex crosswords and the group's performance is high". Notice that by taking advantage of the possibility of defining disjunctions within the rules and being able to combine such rules sequentially, it is possible to obtain a great wealth of possible lessons' evolutions. Finally, since some of these rules may contain conditions which concern the user model, not all of them are applicable (e.g., in the above example, in case the group's current performance is low, only the simple crosswords is proposed), resulting in an overall network which is always compatible with the current users' profiles.

Broadly speaking, the teacher loads the chosen lesson from a database resulting in the construction of an initial event network (i.e., the set of events, positioned over time, communicated to users like videos, text messages, questions, etc.). The network is, through the application of the rules, afterwards customized to the users participating in the lessons. By executing the lesson, then, events, representing stimuli and requests, are dispatched, at proper time, to the users associated to them. It is worth noticing that in order to foster interaction and collaboration among them, the distributed information may be partial, requiring users the need to send messages to other students so that they can build an overview and respond appropriately to the challenges posed by the system fostering, thus, cooperation. Whenever the profile of a user changes because, for example, his/her level of fatigue increases, specific rules are
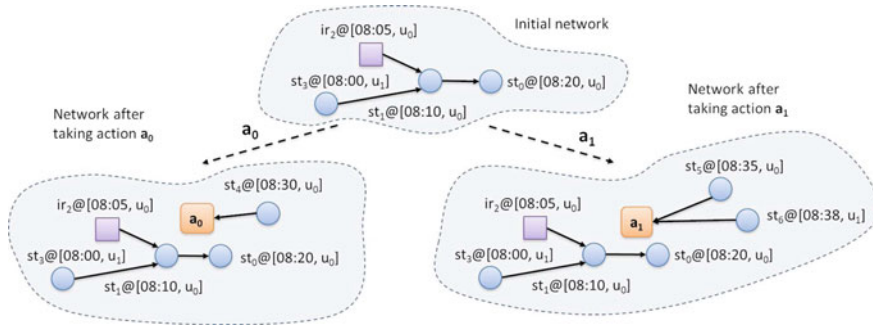
Fig. 3 An example of *LECTurE* dynamic lesson adaptation in case either action a_0 or action a_1 is performed by a user

applied resulting in an update of the network so as to bring it back to be "consistent" with the current status of the users. Similarly, updates might happen as a consequence of the users' actions, resulting in a network which is always representative of the initial lesson while being dynamically adapted to the specific context.

As an example, Fig. 3 shows, at its top, an initial network containing three stimuli (i.e., st_0, st_1 and st_3) and an interaction request (i.e., ir_2) representing, for example, a question. Each event has its own execution time and its target user (e.g., st_0 will be dispatched at 08:20 to the user u_0). In the figure, the arrows represent the causal relations among the events that emerge from the application of the rules. In other words, the st_1 event is in the network "because of" the st_0 event. Another way of looking at it, from a causal point of view, is to consider the st_1 event as a "condition" for the existence of the st_0 event. Finally, notice that in order to simplify the explanation and make the speech clearer we have omitted the information regarding the temporal constraints which, however, can be considered embedded within the arrows.

At 08:05, the interaction request ir_2 requires the user u_0 to take some action (i.e., either a_0 or a_1). Suppose, as an example, that the action a_0 is performed, the network is updated to the one on the left by adding the new stimuli st_4 at 08:30 for user u_0. Again, the st_4 stimuli is added within the network "because of" the a_0 action. Conversely, in case action a_1 is performed, the network is adapted to the one on the right by adding the new stimuli st_5 at 08:35, for user u_0, and the new stimuli st_6 at 08:38, for user u_1. As already mentioned, it is worth noticing that this type of adaptation occurs in a similar way as a consequence of the actions carried out by the users as well as for the dynamic changes that occur to their profile, ensuring a discrete availability of flexibility and adaptation to the particular conditions that may arise in the different lessons.
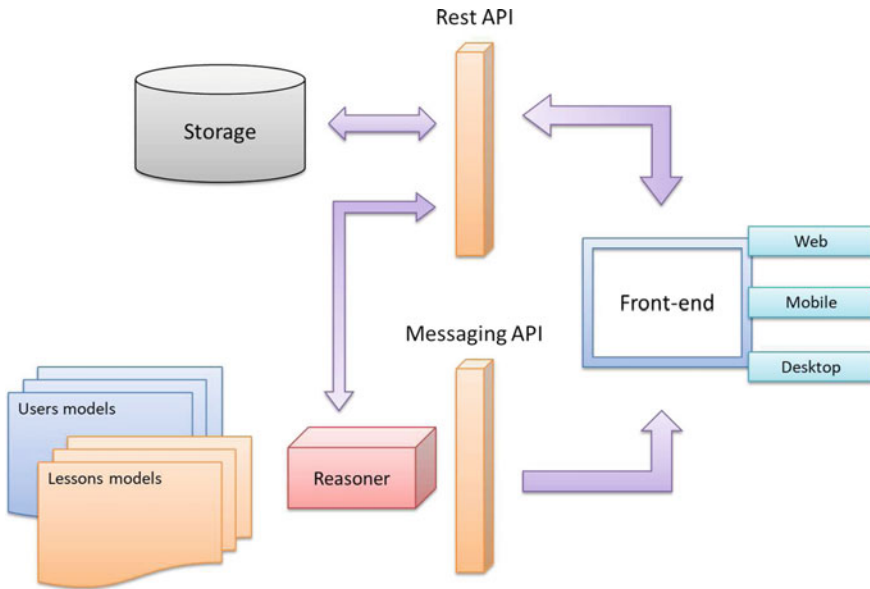
Fig. 4 The *LECTurE* main modules

4 The System Architecture

This section briefly introduces the *LECTurE*[3] system software architecture. In particular, Fig. 4 sketches the modules described earlier, introducing some standard components, responsible for the communication among the different modules, which act as a glue.

Specifically, the architecture contains a module for the permanent storing of relevant information. A reasoner, constituted by a timeline-based planner [5], works to orchestrate the different models, specified by means of first-order rules, of the users and of the lessons. Such rules, in particular, are responsible for fulfilling the updates to the network as described in Sect. 3.2. A front-end module deals with providing (web, mobile, and/or desktop) graphical interfaces to the users. Finally, it is worth noticing that the communication among the different modules happens by means of two different software interfaces:

- A REST interface allows the creation, updating, and viewing of permanently stored information. The same interface is used to notify any changes to a user's profile (e.g., the user has moved around a site of historical interest) or possible actions by the user (e.g., answering a multiple-choice question) or, furthermore, messages sent to other users.

---

[3]https://github.com/pstlab/LECTurE.

- A messaging interface, made through WebSocket and/or MQTT, allows to send different types of messages to the users. A typical example of a message is a stimulus like, for example, an interaction request consisting of a multiple response question, a questionnaire, a map, a question for detecting the psychological state of the user, etc.

5   An example of *LECTurE* at work

This section introduces a concrete example intended to show the *LECTurE* system at work. As already mentioned, we aim at fostering physical and cognitive activity while reducing social isolation of elders. In addition, we always keep in mind we are addressing a segment of the population subject to social disadvantage due to their physical condition. To achieve this we can exploit some architectural elements of Rome. One of the characteristics of Rome, indeed, consists in the presence, on walls or on the corners of historic buildings, of a number of Madonnas painted under canopies. The idea consists in sending stimuli to users so as to guide them to visit such shrines, customizing the path to their psychophysiological state. By exploiting georeferencing, the *LECTurE* system asks users to take pictures of the shrines to be used, afterwards, to build games which, played together, reduce social isolation and stimulate cognitive activity. It is worth noticing the similarity of this scenario with the activities currently carried out by the Televita association. In particular, visiting the shrines mimics the *Art History lessons and cultural events* activity. Analogously, the automatically built games are intended to reinforce the *AttivaMente laboratory*.

The first step for the implementation of a scenario of this kind consists in collecting information about the shrines like, for example, a brief history about each of them and their GPS coordinates. Additionally, a customized questionnaire, administered to users, is intended to extract an initial profile to be used as a baseline. This information is sufficient to generate a series of paths to which we can associate different difficulty levels measurable in terms of the path length. The different paths can be computed, in a pre-processing step, by solving a top-*k* Traveling Salesman Problem (TSP) (where *k* is the number of shrines which are planned to be visited) and encoded into rules. Additionally, the number of shrines can be reduced by filtering out those that are too far from a certain point which is considered as the starting point of the excursion (e.g., consider only those shrines which are closer than 3 km).

Starting by the initial profile, a path, compatible with the profile, is selected like, for example, the dashed one in the center part of Fig. 5. The path is then, step by step and at proper time, suggested to the user by means of customized stimuli (e.g., "go to pos1", "the shrine at pos1 has been built in 1796", "take a picture at the shrine in pos1", "go to pos4", etc.). Additionally, by taking into account physiological data, the system can adapt the route switching, for example, once the *LECTurE* system realizes that the user is not too tired, to the solid one, fostering physical activity and a prolonged interaction with the other involved users. Finally, once back from the trip, the *LECTurE* system builds a "memory" game, challenging the participants to
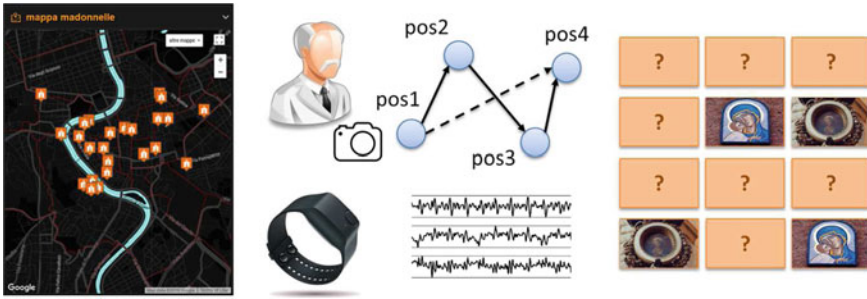
Fig. 5 The *LECTurE* system at work

discover, in the fewest possible steps, hence stimulating cognitive activity, pairs of the taken pictures hidden under the tiles.

6 Conclusions and Future Works

This paper introduces the *LECTurE* system as an AI-based Intelligent Tutoring System specialized for supporting active aging. The paper defines the *LECTurE*'s initial functionalities, the architectural entities and a specification of the technological components. We have introduced the general idea of supporting older people in maintaining themselves mentally and physically active being helped by some intelligent technology both during a class and during excursions. The ICT intelligent core makes use of a specific kind of artificial intelligence technology, called automated planning, which is responsible for orchestrating the different modules of the *LECTurE* system, resulting in the creation of a baseline "lesson" which is dynamically adapted to the actions taken by, as well as to profile updates of, the users, so as to integrate both personalized stimuli and requests to the users over time. Although the application is currently under development, the constant contact with Televita's volunteers is allowing the transition from a lab prototype to an incrementally more robust version of the system to be tested in realistic scenarios.

References

1. Bacon L, Cesta A, Coraci L, Cortellessa G, De Benedictis R, Grilli S, Polutnik J, Strickland K (2012) Training crisis managers with PANDORA. In: Proceedings of the 20th European conference on artificial intelligence, pp 1001–1002. IOS Press
2. Cesta A, Cortellessa G, Benedictis RD (2014) Training for crisis decision making—an approach based on plan adaptation. Knowl Based Syst 58:98–112

3. Cesta A, Cortellessa G, De Benedictis R, Fracasso F (2017) A tool for managing elderly volunteering activities in small organizations. In: Esposito F, Basili R, Ferilli S, Lisi FA (eds) AI*IA 2017: advances in artificial intelligence

4. Cortellessa G, Fracasso F, Sorrentino A, Orlandini A, Bernardi G, Coraci L, De Benedictis R, Cesta A (2017) ROBIN, a telepresence robot to support older users monitoring and social inclusion: development and evaluation. Telemed e-Health

5. De Benedictis R, Cesta A (2015) Integrating logic and constraint reasoning in a timeline-based planner. In: AI*IA 2015— XIVth international conference of the Italian association for artificial intelligence

6. Ghallab M, Nau D, Traverso P (2004) Automated planning: theory and practice. Morgan Kaufmann Publishers Inc

7. Gronlund NE, Brookhart SM (2009) Gronlund's writing instructional objectives. Pearson/Merrill Prentice Hall

8. Muscettola N (1994) HSTS: integrating planning and scheduling. In: Zweben M, Fox MS (edS) Intelligent scheduling. Morgan Kauffmann

9. Ohlsson S (May 1986) Some principles of intelligent tutoring. Instr Sci 14(3):293–326. https://doi.org/10.1007/BF00051825

10. Wenger E (1987) Artificial intelligence and tutoring systems: computational and cognitive approaches to the communication of knowledge. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA