AAL PROGRAMME	**PersonAAL**	

Deliverable 1.2.b

Monitoring and Behavior Analysis (Revised)

Responsible Unit:	**CNR-ISTI, HIIS Laboratory**
Contributors:	**FCID**

Document Technical Details:

Document Number	D1.2.b
Document Title	Monitoring and Behavior Analysis (Revised)
Version	1.0
Status	Final
Work Package	WP1
Deliverable Type	Report
Contractual Date of delivery	2017-09-30
Actual Date of Delivery	2017-09-29
Responsible Unit	CNR
Contributors	FCID
Keywords List	Older adults, Contextual parameters, Behavior, Analysis, E-Health, Devices
Dissemination Level	Public

Document Change Log:

Version	Date	Status	Author	Description
0.1	21-09-2017	Draft	CNR (M.Manca, P.Parvin, F.Paternò, C.Santoro)	First draft
0.2	26-09-2017	Reviewed draft	FCID	
1.0	29-09-2017	Final version	CNR (M.Manca, P.Parvin, F.Paternò, C.Santoro)	Final Version

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

Executive Summary

Current technology offers the opportunity to detect a wide range of environmental and user-related parameters. Monitoring such parameters allows us to build knowledge about the context around the elderly. However, tools able to integrate in real time data coming from all the potential sensors are missing, as well as tools for analyzing such data and acting consequently when certain conditions are met. In this deliverable we describe the latest refinements that have been done to the PersonAAL architectural module devoted to "Monitoring and Behavior Analysis", which aims to detect whether there are discrepancies between the elderly's expected behavior and the actual one in AAL scenarios. Our solution exploits task model specifications in which the user's expected behaviour is represented, and compares such behaviour with sequences of events logged in the current context, which should provide information about the actual user behaviour. By using associations between elementary tasks defined in the considered task model and events gathered in the current context, the system is able to detect possible deviations and, depending on the type of deviation identified, trigger accordingly suitable actions addressing the concerned anomaly.

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

FCT Fundação para a Ciência e a Tecnologia
Forskningsrådet
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Contents

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

1 INTRODUCTION

The rapid ageing of population is at the same time an extremely good achievement and a critical global challenge, because in the future it will be difficult to ensure a good quality of life of an unprecedentedly large elderly population in a sustainable manner. Nowadays, there is an increasing need for supporting the elderly in preserving an independent and healthy lifestyle in their homes rather than through more expensive hospitalization solutions. In such scenarios, one aspect that is getting increasing importance is monitoring user's daily living activities, i.e. routines on which the ability of a person to live independently is typically assessed (e.g. bathing/showering, dressing, food preparation, eating, personal hygiene, taking medicines, telephoning). While there are several contributions addressing the issue of activity or routine recognition in smart homes, relatively fewer solutions have addressed the challenging topic of detecting "unusual" behaviour on the user's side, i.e. possible deviations from expected users' routines. This topic is highly critical for the deployment of remote monitoring systems for elderly, because changes in the elderly's routines can be a potential sign of elderly's changing health conditions due to, e.g. initial signs of decline, beginning of unhealthy habits, and modifications of user's abilities.

The identification of anomalous patterns can be highly valuable for experts to make decisions in emergency or risky situations or can be used to provide the elderly themselves with some relevant information and notifications helping them to properly carry out their own activities. Anomalies could affect various routines' aspects, e.g. the order according to which activities are carried out, the place where the activity takes place (e.g. sleeping in the bed vs. on the sofa), the time when the activity started (e.g. too early/too late) and duration (e.g. too long/too short) of the activity compared to the expected one. Furthermore, anomalies can have various degrees of severity (e.g. the consequence of not taking a pill could be more severe than just forgetting to take a shower). In addition, they can be detected by considering short timeframe monitoring periods (e.g. if detected data suddenly present values that are significantly different from the expected ones), or they could require a longer monitoring timespan in order to be identified as an actual risky situation (e.g. in case of more gradual changes in user behaviour).

In this deliverable, we present an approach to detecting abnormal behaviour due to significant changes in user behaviour in AAL scenarios. Our solution exploits task model specifications in which the user's expected behaviour is represented, and compares such behaviour with sequences of events logged in the current context, which should provide information about the actual user behaviour. By using associations between elementary tasks defined in the considered task model and events gathered in the current context, the system is able to detect possible deviations and, depending on the type of deviation identified, trigger accordingly suitable actions addressing the concerned anomaly.

A first version of the architectural module "Monitoring and Behaviour Analysis" has been already described in D1.2.a. This deliverable describes the latest extensions and refinements that we have done on this module of the PersonAAL platform.

In particular, this document is structured as follows: after the Introduction (Section 1), Section 2 is dedicated to the monitoring and behaviour analysis support, describing the methodology as

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

FCT Fundação para a Ciência e a Tecnologia
Forskningsrådet
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

well as the architecture specification of the solution, also providing an example application. Finally, we conclude with some remarks and plans for future research.

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

2 MONITORING AND BEHAVIOUR ANALYSIS

Behavior analysis is devoted to the detection of patterns in a set of data gathered from sensors, which do not conform to the expected behavior. The aim is to identify possible anomalous situations in elderly's routines, which may indicate initial signs of decline. Our approach is based on the *comparison* between *the actual behavior* of the elderly with the *expected one*. The actual behavior is derived from the analysis of the data gathered, monitored and made available by the Context Manager, and it is aimed to infer/recognise the actual user activities. The expected behavior is specified in a task model, built with the collaboration of caretakers, and in which the typical routines of the considered elderly are modelled. By comparing the two sets of data, we plan to derive useful information about how compliant the elderly behavior is with her expected routines and whether possible actions are needed to cope with the identified deviation. Examples of abnormal behavior in a AAL scenario could be the following ones: prescribed medicine is not taken in due time, a non-prescribed medicine is taken, elderly sleeps at unusual time (e.g. after breakfast), elderly goes to sleep without having dinner, abnormal sleep patterns (e.g. sleep less/oversleep), abnormal activity patterns (e.g. no physical exercise or exercise less than usual).

2.1 Gathering and Monitoring Contextual Data

The monitoring of the contextual parameters is carried out by the Context Manager. The main objectives of this subsystem are detecting the current values of a wide range of variables (related to user, environment, technology, etc.) and informing other architectural modules about relevant changes in such values.

The Context Manager is composed of a Context Server and a set of Context Delegates. The Context Server contains all the functionalities for storing and analyzing data in a centralized manner, while the Context Delegates are distributed in the user device(s) or in the environment. A context delegate is a software devoted module to get data from a sensor (a real one or a 'virtual' one, i.e. an external software service able to gather some data, such as weather forecast service), elaborate them and send them to the Context Server. For example, a Context Delegate for the sensors of the Bitalino chest band built by the Plux partner has been developed.

The data sensed by the Bitalino chest band are gathered from the various sensors included in it (e.g. accelerometer, ECG, temperature,..) and include the respiration and heart rate, the body temperature in degrees, the number of steps performed from the previous transmission and the list of the body positions (among standing, supine, prone, right and left) detected in the time interval. For convenience purposes, in Figure 1 we just provide a graphical overview of the Context Manager. However, detailed information about how this module works, including the specific mechanisms used for data retrieval, collection and organization, as well as the ones used for communicating with it (e.g. update context data by the Context Delegates, subscribe to the Context Manager and being notified by it) is more extensively provided in D1.2.a (Monitoring and Behavior Analysis – Initial) and also in D1.1.b (Architecture Specification -Revised).

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

FCT Fundação para a Ciência e a Tecnologia · Forskningsrådet · Schweizerische Eidgenossenschaft / Confédération suisse / Confederazione Svizzera / Confederaziun svizra

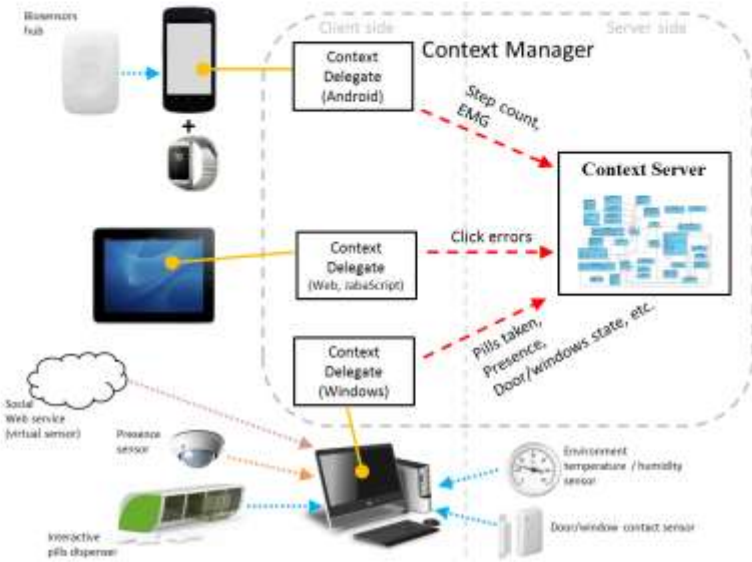

Figure 1: An overview of the Context manager

2.2 Specifying the Expected Behaviour in Task Models

Task models indicate the logical activities that should be supported in order to reach users' goals (they are typically used when designing interactive applications). In our case, we plan to use task models to describe the elderly's **expected** behavior (in terms of, e.g. specific tasks/activities to carry out, typical daily routines followed, particular requirements and needs the elderly can have, which could be connected with their current health state or disability/disease the user can suffer from). The task model is then provided as input to the Behavior Analysis module which compares its data with the information gathered by the Context Manager and describing the **actual** behavior of the user. Task models should be created by designers/developers with the collaboration of caregivers (or even the elderly themselves), who have an intimate knowledge of the needs of their care beneficiaries. In order to specify task models, we use a notation developed at CNR-ISTI, the ConcurTaskTrees (CTT) notation (Paternò, 2000) which models the activities in a hierarchical manner, and support the specification of a set of relationships between them (so-called "temporal operators"). The main ones (with the associated symbol) are the following ones:

- Choice: []
- Order Independence: |=|
- Interleaving: |||
- Disabling [>

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

- Enabling >>
- Suspend/Resume |>
- Iteration *

As for the underlying environment, ConcurTaskTrees Environment (CTTE) (Mori et al., 2002), is the tool (publicly available at http://giove.isti.cnr.it/ctte.html) supporting the interactive creation and simulation of CTT task trees.

2.3 The Method for Identifying Deviations in Elderly's Behaviour

Our method is composed of several steps. The first steps are aimed to configure and prepare all the necessary data needed for the analysis and connections among them. Then, we have the steps performed at runtime, while the elderly are actually monitored in their daily behavior.

Set up a suitable task model for the analysis.
In this step, a CTT task model describing the planned activities from the user's viewpoint should be created with the help of relevant stakeholders (e.g. elderly's caregivers). However, to be exploited by the automatic Behavior Analysis, the task model should be checked for its correctness by relevant stakeholders. In addition, since a task model can include activities of various categories, it could be subject to another step consisting of *pruning* categories of tasks that are not relevant for the automatic analysis. Indeed, the CTT notation allows designers to indicate how the performance of the task should be allocated according to these categories: *application* (when the task is carried out by the system), *interaction* (when the task is carried out through an interaction between the user and the application), *user* (the task represents a cognitive task, so entirely carried out by the user), *abstraction* (when the task allocation is a combination of different types of allocation). As such, only tasks that can be detected through sensors and/or user interactions can be considered in the analysis, so the tasks that are associated with user's internal cognitive activities ("user" tasks) are automatically pruned from the task model and excluded from the analysis.
It is also worth pointing out that in this work we consider not only users' common daily living activities but also user's access to services and user interfaces through digital devices and appliances available at home. Thus, actions implying an on-purpose interaction with any physical or digital artefacts, as well as with the environment will be classified as "interaction" tasks, while application tasks refer to either feedback from the application or to application actions that occur without being an effect of a specific user interaction.

Associate events occurred in the context with elementary tasks in the task model.
In this step, we need to associate each elementary task in the pruned task model to one or more events gathered in the context. This association allows for using the semantic information contained in task models to analyze the sequence of corresponding associated events: when the event(s) associated with a task occurs, then the corresponding task is considered completed. However, it is worth noting that tasks could not have the same granularity of events. In some cases, there is a 1:1 mapping between tasks and events (e.g. the "user switches on the lights" task corresponds to the event: "light switched on"). In other cases, multiple events can be associated with a single elementary task (N:1), thus a complex event expression is needed. For instance, if we consider the "sleeping" task, then a corresponding event expression (depending

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

on the available sensors) could be: (user posture= lying down AND user's heart rate= low AND user's bed pressure sensor=on AND user motion=none). In case of a complex event expression the task is interpreted as actually completed only when all the associated events or conditions have been detected in the current context. In Figure 2 you can find a screenshot of the Association Tool, used to build the so-called "association file", containing the mappings between the elementary tasks of the task model considered for the deviation analysis (and shown in the left part of the tool), and the associated events detected in the current context (Figure 2, right part). In the shown example the task "Taking medicine 1" (included in the considered task model) is associated with a trigger involving the state of "Pill Dispenser 1" (the state of this smart object is automatically detected in the current context). The idea is that, if the "Pill Dispenser 1" is open, the task "Taking medicine 1" has been carried out.

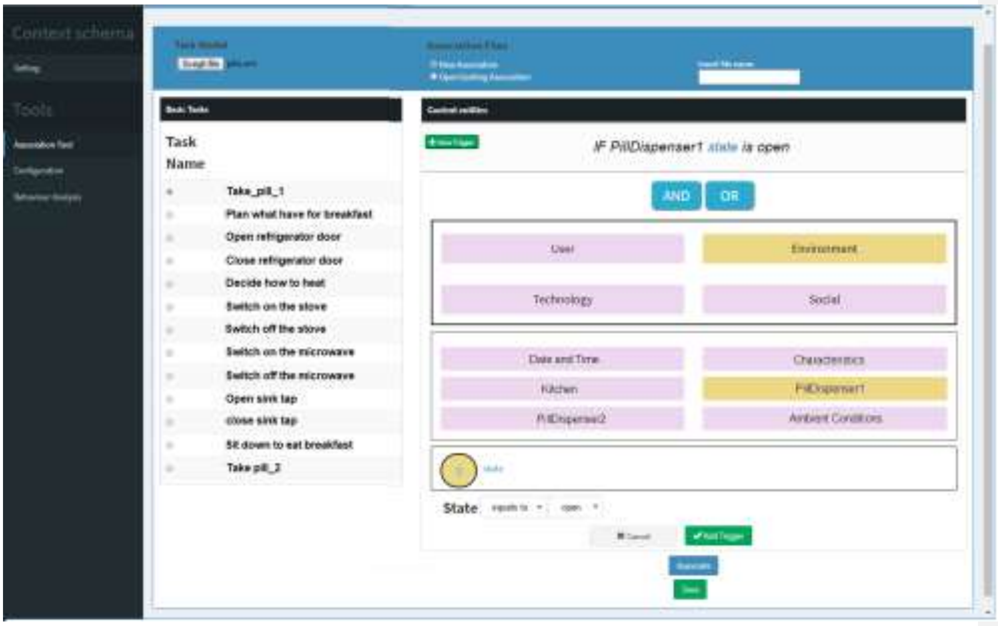

Figure 2: Association Tool

Definition of the actions to perform when the deviation is detected

Different types of actions can be triggered depending on the deviation detected. Currently we are considering: modify the user interface (presentation / content / navigation); activate some functionality; Generate alarms (to highlight some potentially dangerous situations); Send

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

reminders (to indicate a task that should have been accomplished); Provide persuasive suggestions (to encourage users to change their behavior); Send prompt messages; Provide explanation messages; Change the state of an appliance (light, fridge, …)

Log relevant contextual events. The goal of this step is to log the events occurring in the context where the elderly actually live, also associating them with a timestamp needed for further analysis (e.g. precisely identifying the beginning of an activity, calculate how much time the user spent doing a specific task). As we will see in the next section describing the architecture of the Behavior Analysis module, there is a Deviation Analysis (sub-)module, which subscribes to the Context Manager to receive such events, and then save them in a specific data structure for further analysis. It is worth pointing out that the events considered by the Deviation Analysis module will not be all the possible events that could be gathered in the current context, but only the ones which are associated with the elementary tasks contained in the task model.

Compare the actual user behavior with the expected behavior specified in the task model and identify possible deviations.
This comparison consists of checking whether the sequence of basic tasks associated with the sensed events satisfies the temporal constraints specified in the task model. This means comparing the set of tasks enabled in the task model (according to the current execution state of the model) and the task(s) associated with the real events occurred in the context: if the latter ones do not belong to the first one, then a deviation occurs.
Several types of deviations can be identified as a result of this comparison. In particular, since user's activities can be described in terms of a set of task attributes (such as temporal relationships, location, time, resources handled), it is possible to identify a number of corresponding types of task-related anomalies affecting elderly people. Some examples include i) unusually long inactivity: when a task has a temporal relationship attribute and no other tasks has been executed after that for a long period (e.g. because the elderly has fallen and has lost consciousness); ii) unusually short activity which equally refers to the temporal dimension (e.g. the lunch was too short); iii) unusually frequent activities when a task is not marked as iterative but it is performed repeatedly (e.g. too many visits to the toilet); iv) violations of task order relationship (taking medicine before having a meal instead after that); v) wrong object/resource handled (e.g. the elderly unintentionally took a medicine not prescribed instead of the right one); vi) missing task (the user goes to sleep without having dinner or the user does not do exercise).

To sum up, main anomalies occurring on tasks can be categorized according to the following three main types:

- *Less*: a task that was expected to be carried out, has not been performed;
- *More*: more tasks than expected have been performed;
- *Difference*: expected tasks have been performed but in a wrong manner (in terms of e.g. temporal order, time, location, objects manipulated) from the planned one;

A more extended classification of such task-related anomalies is provided in the next subsection 2.3.1. Each type of deviation can also be *critical* or *uncritical* depending on the attribute criticality level associated to the task(s) involved in the deviation.

Analyze the deviation and identify a suitable action for addressing it. When a deviation is detected, the system analyses the type of deviation (in terms of, e.g. distance from the expected behavior, criticality level of involved task) and identifies suitable remedial actions to take (e.g. generate alarms or reminders to the elderly) in order to cope with the identified anomaly, even by escalating them if no reaction from the elderly is obtained, e.g. sending an alarm message also to the caregiver if no action is obtained in reaction to an alarm previously sent to the elderly. The possible actions to take are identified in a semi-automatic manner, among the ones that are supported by the considered application. Depending on the current context, the notification to be sent to the user can be adapted by considering various factors, for instance, how to render it depending on the current contextual situations (for instance, using a vocal message if the users are far from the device in order to get their attention).

Commented [CAPdAD1]: A more detailled explanation should be provided. The possible actions are provided by the stakeholders/caretakers or is it a set of actions that are applicable to all tasks/deviations? How does the identification process works? Is it automatic? Is it based on an attribute of the action and, if so, which attribute?

2.3.1 Possible Task-related Anomalies in Elderly Behaviour

As already mentioned, an anomalous situation is characterized by a difference between the observed situation (derived from sensing the current user context) and the expected one (described in a task model). The identified deviation could be characterized by different degrees of severity and could be the sign of various situations: the beginning of an unhealthy habit of the elderly, the insurgence of an illness, or even a serious physical/mental decline. Thus, analyzing the behavior of the elderly can be useful to highlight trends as well as for giving recommendations to the elderly, sending them health-related reminders, etc.
We provided a systematic categorisation of the various types of deviations that can affect a task, which we have associated with specific keywords:

- **None**. The task has not been performed or has been performed without producing any result. This deviation is decomposed into:
  - **No input**. Lack of initial information necessary to perform a task;
  - **No performance.** The task has not been performed;
  - **No output.** The task has been performed but its results are lost;
- **Other than**. The tasks considered have been performed differently from what the designer intended in the task model. This deviation is decomposed into:
  - **Less.** It can be further decomposed into *less input*, *less performance*, *less output* according to the cause of the deviation;
  - **More**. It can be further decomposed into *more input*, *more performance*, *more output* according to the cause of the deviation;
  - **Different**. It can be further decomposed into *different input*, *different performance*, *different output* according to the cause of the deviation;
- **Ill-timed**. The tasks considered have been performed at the wrong time. This deviation is decomposed into:
  - **Time**. It can be further decomposed into *too early* and *too late* according to when the activity occurs (respectively earlier/later than when was planned);
  - **Duration.** It can be further decomposed into *too long* and *too short* according to the duration of the activity;
- **Ill-located**. The tasks considered have been performed in the wrong location

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

European Commission · Ministero dell'Istruzione, dell'Università e della Ricerca · FCT Fundação para a Ciência e a Tecnologia · Forskningsrådet · Schweizerische Eidgenossenschaft / Confédération suisse / Confederazione Svizzera / Confederaziun svizra

2.4 The Architecture of the Solution

Figure 3 shows the architecture of the solution, which we designed and developed to support the proposed method. Its behavior can be described according to two main phases: one is devoted to configuring the analysis (e.g. building/selecting the task model, building/selecting the association file), the other one for carrying out the automatic analysis and performing the consequent actions.

In the first phase, the user has to select the task model to use, which specifies how the various activities should be carried out. In addition, using the Association Tool (Figure 2), the user has to build the concerned *association file*, containing the mappings between the elementary tasks of the task model that are of interest for the deviation analysis, and the associated events detected in the current context. The tasks that appear in such a list are basic tasks (see Figure 1 on the left side), tasks that are not further decomposed in the model. As said before, the idea underlying the associations is the fact that the occurrence of a specific event of interest in the current context (they can be interactively selected through the user interface shown in Figure 2, right side) can be translated into the completion of an associated elementary task in the task model.

The second step deals with the detection of possible anomalies in the sensed behavior, and it is centered on the Behavior Analysis module, which is composed of three sub-modules: The Simulator, the Deviation Analysis, and the Action Generator. The basic goal of the Simulator module is to provide the Deviation Analysis with the information needed to decide whether the sequence of events detected by the Context Manager can be translated in a path of elementary tasks that is compliant with the temporal relationships contained in the task model representing the expected behavior. In order to do this, we used the CTTE Task Model Simulator which gets in input i) the specification of a complete and consistent CTT task model, and ii) the next task to perform in the simulation (this task is associated with a relevant event occurred in the context of use). Depending on the task temporal operators, the Simulator calculates the updated list of the tasks currently enabled after the execution of the given task to perform in the current state of the simulation, and returns such updated list to the Deviation Analysis module.

The Deviation Analysis module first subscribes to the Context Manager in order to be notified when the events specified in the association file occur in the context. After having subscribed to the Context Manager, the Deviation Analysis receives relevant events from it and, according to them, it is able to check whether any deviation occurs in the received sequence of events. In order to do this, the Deviation Analysis module gets in input three elements: i) the association file; ii) the enabled task sets (from the simulator); iii) the event lastly occurred in the current context (from the Context Manager).

Thus, the behavior of the Deviation Analysis is the following one: it takes the event just occurred in the context (and provided by the Context Manager), retrieves in the association file the corresponding elementary task, and checks whether such task belongs to the set of enabled tasks (provided by the Simulator module). If it is the case, it means that the received event was expected according to the task model: in this case, the process iterates in a similar manner for the next received event. Otherwise, i.e. if the expected action is not received within a fixed time interval (specific temporal thresholds can be set up for this purpose), the Deviation Analysis

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

FCT Fundação para a Ciência e a Tecnologia
Forskningsrådet
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

module signals to the Action Generator module the deviation that occurred in the detected behavior.

Depending on the type of deviation identified, the critical level of the involved task, and taking into account other contextual aspects (for instance: possible user preferences, in order to identify the best action to do according to what the user prefers; the current user location and device used, in order to be sure that the user is actually aware of e.g. an incoming message), the Action Generator module selects the best action to do for addressing the identified anomaly. For instance, if the deviation is of type *Less* and it involves a critical task, then the Action Generator could decide that an alarm should be sent to relevant stakeholders. In particular, it generates a high-level action (specified according to a specific XML-based language) to be sent to the considered web application (for instance this could be the Remote Assistance application developed by Reply within the project), which interprets it and performs accordingly the requested UI changes (e.g. modify the font size of some elements to make them more readable) or activate some functionality (e.g. generate an alarm) to deal with the identified deviation.

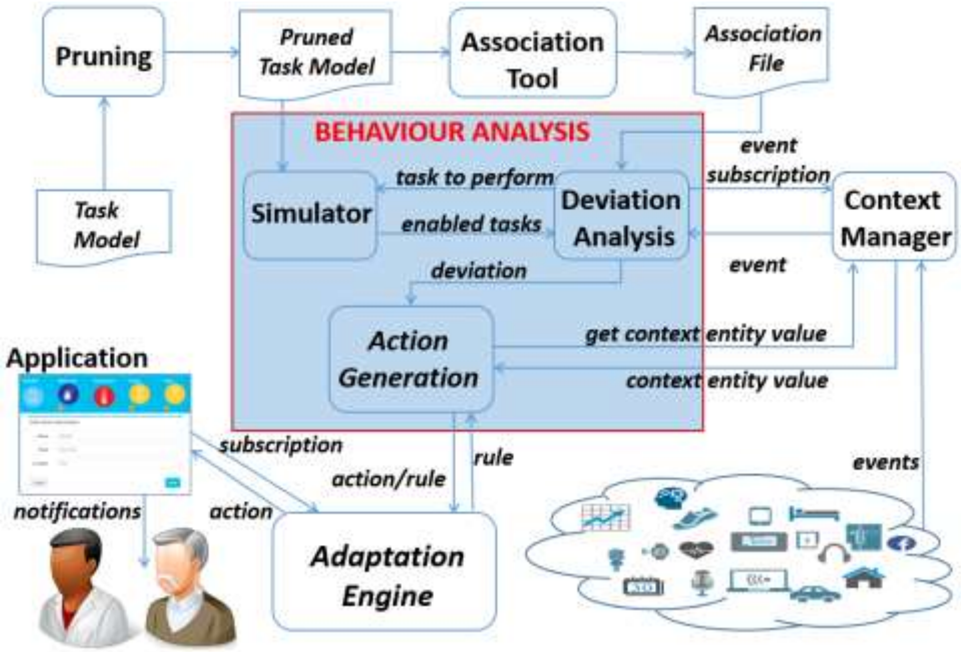

Figure 3: The Architecture of the Solution

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

FCT Fundação para a Ciência e a Tecnologia
Forskningsrådet
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

The Action Generation module, in addition to the identified deviation, also receives from the Adaptation Engine some rules, which define the most suitable ways to render the actions identified for addressing the identified deviation by taking into account relevant contextual aspects. For example, if there is a rule such as "If the noise level is less than 50 dB, then sends reminders both textually and vocally", when the Action Generation module receives a notification of a deviation for which a reminder should be sent, then it asks the Context Manager the value of the noise level and depending on the current value, the Action Generator will require the application to render the identified action (e.g. send a reminder) in different modalities (both graphical and vocal if the noise level is low). The rules are specified in a Trigger-Action format (specified according to a XML-based language), and they are supposed to be built with the help of relevant stakeholders (e.g. caregivers). In addition, before starting the Deviation Analysis, the designer has to define the actions that should take place when a deviation occurs. These are actions that can be performed in the considered application or commands to change the state of some appliance (e.g. changing color and intensity of lights).

The application can be any Web application (for project goals, we considered an application supporting elderly remote assistance). When such application is loaded, it has to subscribe to the Adaptation Engine module in order to be notified when a deviation is detected. Moreover, the application must include suitable JavaScript scripts able to interpret the messages received by that module (e.g. the adaptation actions to perform) and apply the required actions on the application elements and structure.

2.5 An Example Application

In the example considered we focus on the task of medicine intake (see Figure 4). The user should take first the Spironolactone pill (used to treat fluid retention in people suffering from heart failures) without food (*pill_1* in Figure 4), then he should have breakfast and, only after that, he should take the Diuretics pill (*pill_2* in Figure 4). Before being used for the actual analysis, all the user tasks (e.g. the task "Plan what to have for breakfast") are filtered out. The tasks "Take pill_1" and "Take pill_2" are associated with the events generated by a Smart Medicine Dispenser which sends to the Context Manager an update every time the elderly takes a pill (we assume that the Smart Dispenser is able to detect the specific pill the user is currently taking). The task "Have breakfast" is split into two sub-tasks: "Prepare Breakfast" and "Eat breakfast". In turn, "Prepare Breakfast" is composed of three sub-tasks that can be executed iteratively in any order: 1) "Take food from refrigerator", 2) "Cook food" and 3) "Use the sink". The first subtask is associated to sensors installed in the refrigerator door, the second subtask can be related to a gas-sensor attached to the stove, while the third one is linked to a sensor connected to the sink tap. Then, the elderly has breakfast (which is detected according to the specific sensors available, e.g. pressure on the chair is detected for a sufficient interval of time).

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

FCT Fundação para a Ciência e a Tecnologia | Forskningsrådet | Schweizerische Eidgenossenschaft Confédération suisse Confederazione Svizzera Confederaziun svizra

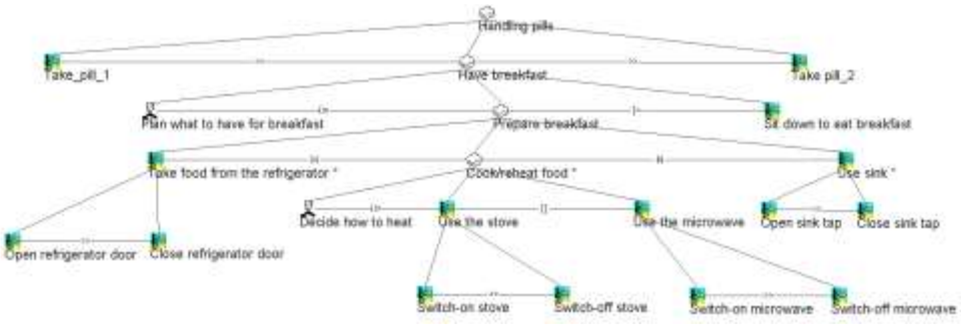

Figure 4: An example

We developed a Web application for caregivers who can monitor the activities of the patients in their daily life. As explained before, at the start up the application subscribes to the Behavior Analysis module in order to be notified when a deviation occurs. If the senior takes the second medicine before having breakfast, the Behavior Analysis module identifies a deviation of type "*Difference*" and then it retrieves the action related to this deviation. The action has been defined by the caregiver and in this case is: *Send an alarm* with the text "*The user should take the Diuretics pill after the breakfast*". The Action Generation module will also consider the rule which takes into account the current noise level in the environment where the caregiver accesses the application; in this case the noise level is quite low therefore the action will be applied in a multimodal way (see Figure 5) in order to better attract her attention.

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

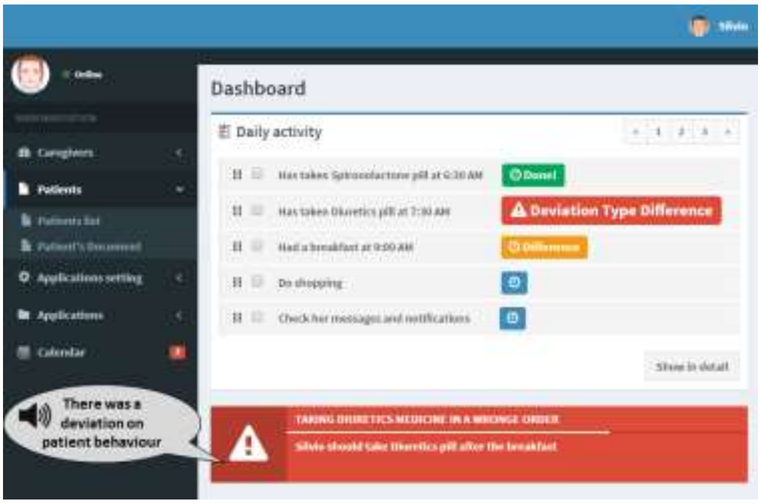

Figure 5: The effect of the action sent to the application after the deviation

In the next example (see Figure 6) we show the results of the analysis by using a tool which allows the user to specify the sequence of events received and, according to the task model considered, delivers the result of the analysis, namely the result of comparing the received sequence of tasks with the information contained in the considered task model. In the example, the task model is still the one shown in Figure 4, and the sequence of tasks associated with events occurred in the current context is the following one (see left part of Figure 6): Take pill 1, Open refrigerator door, Switch on the stove. As the tool correctly shows, in this case, the only deviation identified (till the time when the last action performed occurred in the user's context) was an *omission* (i.e. a task was not done), as the task "Close refrigerator door" has not been carried out.

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

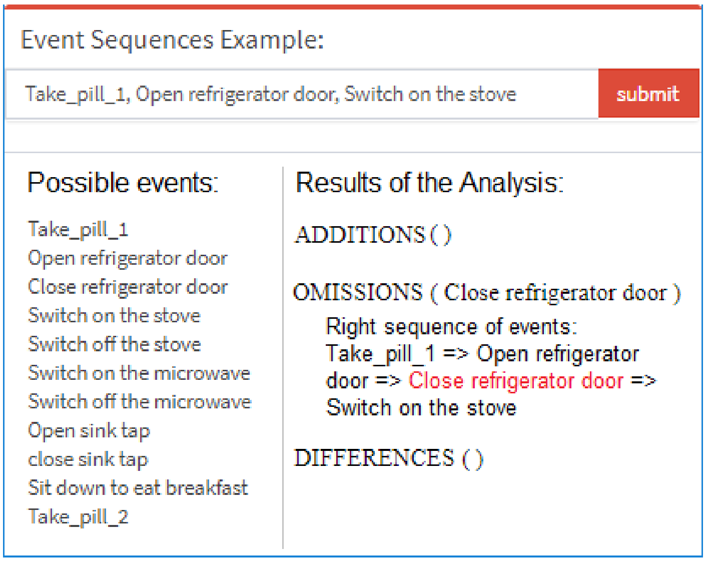

Figure 6: An example of analysis result

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

3 CONCLUSIONS

We have presented a method to detect abnormal elderly behavior, which is important in AAL scenarios because these anomalies may represent early signs of the onset of health-related issues. Our solution exploits (CTT) task model specifications and compares the expected behavior expressed in such models with sequences of events generated by actual user behavior and detected in the current context by the Context Manager. The proposed solution is able to detect significant deviations, and then derive accordingly suitable actions addressing the identified anomaly. We have also shown an example application of such method. Future work will be dedicated to further validation and refinement of the method and integration with the other modules of the PersonAAL platform.

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.

4 REFERENCES

Mori, G., Paternò, F., and Santoro, C., CTTE: *Support for Developing and Analysing Task Models for Interactive System Design*, IEEE Transactions on Software Engineering, pp.797-813, August 2002 (Vol. 28, No. 8), IEEE Press.

Paternò, F., 2000. *Model-Based Design and Evaluation of Interactive Applications*. Springer Verlag.

Serral, E., Valderas, P., Pelechano, V., *Supporting Ambient Assisting Living by using Executable Context-Adaptive Task Models*. International Journal On Advances in Software, volume 7, numbers 1 and 2, 2014, pp.77-87. ISSN: 1942-2628

The project PersonAAL is cofunded by the AAL Joint Programme (AAL-2014) and the following National Authorities and R&D programs in Italy, Portugal, Norway and Switzerland.