****

**Enhanced Affective Wellbeing based on Emotion Technologies for adapting IoT spaces**

**D4.1 User contexts and profiling**

|  |  |
| --- | --- |
| **Consortium:** | **ITEA3** |
| **Deliverable ID:** | **D4.1** |
| **Work package/Task: WP4** |
| **Responsible partner:** | **Arkamys** |
| **Contributing partner(s):** | **ALL PARTNERS** |
| **Dissemination level:** | **Confidential** |

🙔 This page is intentionally left blank. 🙔

Index

 Index

[1. Introduction 6](#_Toc9519824)

[2. Scope and deliverable objectives 6](#_Toc9519825)

[3. Semantic API for Context, behavior and profile 8](#_Toc9519826)

[4. User context 9](#_Toc9519827)

[4.1. Context and behavior as Data structure API 9](#_Toc9519828)

[4.2. Context and behavior through knowledge API 9](#_Toc9519829)

 [Example of the 10](#_Toc9519830)

[4.2.1. Emotion API 10](#_Toc9519834)

[5. User profiling 12](#_Toc9519835)

[5.1. User profile through data structure API 12](#_Toc9519836)

[5.2. User Profile through knowledge API 12](#_Toc9519837)

[6. Use case: Wellbeing coaching for social integration 13](#_Toc9519838)

[1.1.1 Tools for getting user profiles 13](#_Toc9519839)

[1.1.2 Techniques for adapting contents in smart spaces 13](#_Toc9519840)

[1.1.3 Techniques for adapting spaces 13](#_Toc9519841)

[1.1.4 Content and space adaptation techniques implemented by the ITI 14](#_Toc9519842)

[1.1.5 User profile acquisition tool 18](#_Toc9519843)

**Document history**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Glossary**

|  |  |
| --- | --- |
| EmoSpaces | Enhanced Affective Wellbeing based on Emotion Technologies for adapting IoT spaces |
| ITEA3 | Information Technology for European Advancement 3 |

# Introduction

This document is the specification of the user contexts and profiling needed for the Emospaces services such as the sound optimisation system or the wellbeing coaching system.

# Scope and deliverable objectives

The skeleton of the Emospaces architecture is the following

Figure 1 Skeleton of the EmoSpaces solution

The deliverable allows the developing of the EmoSpaces platform API and the EmoService API to guarantee the data exchange between them.

The documents *EmoSpaces\_ITEA3\_Requirements\_D1.2* presents the uses cases of 2 different services:

* Wellbeing coaching:
	+ A1 - Lifestyle coaching to control weight
	+ A2 - Coaching by detection of abnormal behaviors
	+ A3 - Coaching in critical situations



Figure 2 Functional architecture of EmoService A

* Sound Optimisation:
	+ B1 - Context aware sound optimization on home devices
	+ B2 - Context aware sound optimization on wearable device

Figure Functional architecture of EmoService B

Based on these scenario the following chapter aims at defining the data to be accessed form the services.

#  Semantic API for Context, behavior and profile

RDF/S and OWL are Semantic web standard languages, with rich panel of API commonly used to provide developers the possibility to create agents that can handle user preferences and produce contextual knowledge, which are captured either by sensors or directly by humans. However, for the sake of flexibility it is important to leave a certain level of freedom for developers that want to avoid the complexity of OWL since they want represent elementary context knowledge. Therefore, this task has to consider the several meta encoding frameworks dedicated for the web and which can be of particular interest in Emospaces.

The first one is JSON LD, which is a lightweight Linked Data format that can read and written easily by humans as well as software agents. It is based on the already successful JSON format and provides a way to help JSON data interoperate at Web-scale. JSON-LD is an ideal data format for programming environments, REST Web services, and unstructured NoSQL databases.

{

 "@context": "https://json-ld.org/contexts/person.jsonld",

 "@id": "http://emospaces.com/Bob",

 "name": "Bob Sponge",

 "@location": " http://emospaces.com/house/bedroom",

}

JSON-LD Example.

In the same line, RDFa is also an interesting initiative, where it is question of providing a syntax for web content providers allowing them to express RDF structured data in HTML. An important goal of RDFa is to achieve the RDF embedding in HTML without repeating existing HTML content when that content is the structured data.

Let us consider an Emotion Observer in the Emospaces infrastructure, which can deliver emotion of Paola in HTML enhanced with RDF, so this emotion events can be published on het Facebook page, on the home web appliance that displays HTML content on the TV, on the Peper Robot Tablet. Paola like to mark up these events, so that the RDFa-enabled agents can automatically obtain these events and use the content in order to display some indication for Friends or Family Members

RDFa is often more complex than adding a JSON-LD blob into HTML, but there is one important advantage for coupling the metadata in RDFa with the HTML content; is that we can easily and automatically generate the same information in JSON-LD, Turtle, etc, since RDF can be parsed and to any format.

Similar to JSON-LD, RDFa, Micro Data is a form of semantic mark-up designed to describe elements on a web page e.g. review, person, event etc. This mark-up can be combined with typical HTML properties to define each item type through the use of associated attributes. Micro Data has been used by Google, Microsoft and Yahoo! to create the Schema.org as a universally supported vocabulary extension for mark-up languages. It is designed to make the lives of webmasters easier, by offering one standardised mark-up understood by all the major search engines. In fact, there is no best syntax for Schema. The aforementioned encoding frameworks all have advantages and disadvantages. Currently, Schema.org is only fully compatible with Micro Data. A new initiative is JSON Schema that is proposed to Developers providing agents that can produce of hypermedia ready of annotation by using any existing JSON-based HTTP API. JSON Schema documents are identified by URIs, which can be used in HTTP Link headers, and inside JSON Schema documents to allow recursive definitions. Today Google and several companies are actively encouraging to use the schema.org vocabulary.

# User context

This chapter present two API approaches for encoding and managing the User Context and Behavior.

## Context and behavior as Data structure API

|  |  |  |
| --- | --- | --- |
| The first approach the API uses a very simple data structure of the user context, which can be easily implemented in a device with very limited resource, but cannot allow the online e verification of the context and behavior information as it has no formal semantic support. An example of this data structure is given in Table 1.Data Id | Type | Description |
| CTX\_LOC | String | User location : kitchen, bedroom |
| CTX\_ACT | String | User activity : sleep, cooking |
| CTX\_MOOD | String | User mood : happy, sad |
| CTX\_UNKNOW\_USER | Int | Number of unknown user / visitor |
| CTX\_PHYS | String | User physical activity : housekeeping, regular walking |
| CTX\_FALL | Boolean | Detection of user falling |
| CTX\_LYING | Boolean | Detection of user lying |

## Context and behavior through knowledge API

In the second approach, the use of Emospaces ontology will enable also Agents to better sharing knowledge about the User Context, but also Behaviour and Preferences and their interoperability with the different reasoning systems proposed in the task xxxx.

The Emospaces API will use the Emospaces ontology of context to produce high level as well as low level contextual entities. These entities are instances produced by using concepts and relations. Every API can be used to produce an instance type that can be one of the following entities:

• Physical entities describing the user relative or exact location in the indoor space

• Human Activity entities such as any usual activity of daily living.

• Events entities that characterizes changes in any property of the environment space, which has significance or impact on the services delivery adaptation policy.

• Human Emotion entities

Context and Behavior collection by using the Emospaces API must be efficiently improved by using an upper Layer of API that integrates Production Rule Engine. The production API, will combine the context entities (produced from low level context capture agents) with general purpose rules that will allow to generate automatically high level contextual entities. The Smart Rules language developed initially in the SemBySem project and enhanced in ITEA WoO is a good candidate for conceiving context and behavior changes and respond accordingly without user intervention in order to generate additional information. Unlike SWRL but relatively similar to SPIN, Smart Rules can be used to generate contextual entities even in the case of absence of low level context observation, thanks to the negation as failure, which is not supported in SWRL.

### Example of the Emotion API

For handling the emotional Context, an intuitive approach is using the API of EmotionML. This API makes scientific concepts of emotions practically applicable in particular for building Agents that can automatically recognize emotions from sensors, including physiological sensors, speech recordings, facial expressions, etc., as well as from multi-modal combinations of sensors; The production of emotion-related system responses, which may involve reasoning about the emotional implications of events, emotional prosody in synthetic speech, facial expressions and gestures of embodied agents or robots, the choice of music and colors of lighting in a room, etc. The following listings show examples of semantic content that can produced by this API

<emotion category-set="http://www.w3.org/TR/emotion-voc/xml#everyday-categories">

 <category name="satisfied"/>

</emotion>

<emotion dimension-set="http://www.w3.org/TR/emotion-voc/xml#pad-dimensions">

 <dimension name="arousal" value="0.8" confidence="0.9"/>

 <dimension name="pleasure" value="0.6" confidence="0.3"/>

</emotion>

Listing 1. Satisfied Emotion expressed without précising the source

<emotion category-set="http://www.w3.org/TR/emotion-voc/xml#everyday-categories"

 expressed-through="voice">

 <category name="satisfied"/>

</emotion>

….

<emotion expressed-through="posture">

 <info>

 <sensors:sensor idref="camera1"/>

 </info>

 <category name="angry"/>

 </emotion>

Listing 2. Emotions expressed by précising the source as Human Voice or Camera

<emotion category-set="http://www.w3.org/TR/emotion-voc/xml#everyday-categories"

 expressed-through="face voice">

 <category name="satisfied"/>

</emotion>

Listing 3. Multimodal expression of an emotion. The API produce a list of space separated modalities can be indicated in the expressed-through attribute, like in the following example in which the two values "face" and "voice" are used.

The behavior API => To be specified

# User profiling

Similarly, to Context, Defining User profiles to handle their preferences can be done with two approaches.

## User profile through data structure API

In the first approach, the developer can use very basic structure to handle the user profile such as in the example given in table 2

|  |  |  |
| --- | --- | --- |
| Data Id | Type | Description |
| PROF\_FIRST\_NAME | String | User first name |
| PROF\_LAST\_NAME | String | User last name |
| PROF\_ADDR | String [] | User address |
| PROF\_EMAIL | String | User email |
| PROF\_PHONE | Int | User phone number |
| PROF\_BIRTH | Int [] | User birthday ( used to compute user age) |
| PROF\_HEALTH | String | User health condition |
|  |  |  |
| PROF\_HEAR | Int [] | User hearing profile |

## User Profile through knowledge API

In the second approach, the challenge is to provide API to build configuration GUI enabling the users to easily express or value the quality of experience QoE corresponding to how the EMospaces services are consumed and how they have been successfully adapted to their preferences. In some words, what are the key parameters describing the quality of experience and preferences of users in Emospaces. Note that QoE and Preferences are strongly related.

Once these parameters identified we need to solve an important issue, which is the uncertainty since users are subjective in their evaluations as well as in the way they define their quality of experience. The latter is dependent to the context and the Quality of the Emospaces Services (QoS).

However, in the context of Emospaces application it is important that the user profile management API allows the consideration of the uncertainty on user preferences. The uncertainty can be considered under the following aspects:

• Vagueness: When the user preference over a specific QoS attribute is difficult to quantify precisely, it becomes nontrivial to decide on an appropriate weight for the attribute.

• Inaccuracy: The aforementioned motivating scenario shows that an inexperienced user with no cognition about what the realistic QoS values are may pick 100% availability as the only preference for selecting a responsive service that is available 24x7. The Emospaces services cannot be ranked satisfactorily for the user due to the user-provided incorrect user preference specification.

• Incompleteness: User satisfaction with a given Emospaces service could be impacted negatively when some important QoS attribute is not specified, which could be caused by the user’s lack of service usage experience.

# Use case: Wellbeing coaching for social integration

### Tools for getting user profiles

In this use case the detection of the user emotional state and their level of stress is carried out through the platform developed by ERL. A camera detects the emotional state of the user at the beginning of training, and depending on this, the specialist can set up the scene. Once the training scene has started the user stress level is detected with the gyroscope. The ERL platform is used again at the end of the test to measure the emotional state when the training ends.

### Techniques for adapting contents in smart spaces

The Virtual Reality simulation of this use case presents unusual situations or with a goal of cognitive stimulation of the user. These situations, represented as virtual scenes, are set up before the user mood and depending on previous historical uses. In this way scenes are configured with different values for lighting, sound, waiting time or the gender of the main character who appear in the pictograms.

### Techniques for adapting spaces

In our case of use, once the simulation is running, the user's mood and stress state is detected and processed. The scene adapts its contents according to that state, for a better experience and the achievement of the objectives of the training in a satisfactory way. This adaptation is carried out mainly in the exterior and interior lighting of the scene, the modification of environmental sounds and music to reduce the level of stress.

As it is a virtual simulation in which the user must perform some simple tasks, we need they to be aware of some instructions, so a state of high excitement or excessive relaxation can undermine the training objective of the simulation. To solve this, the software will react by modifying the parameters described above in order to attract the user's attention.

### Content and space adaptation techniques implemented by the ITI

From data representative of the present use case, analyses have been carried out for predictive purposes both to inform the experts directly during a session, and to be applied in a hypothetical automatic reactive data flow, as detailed in deliverable 2.4. The data used in the experiments explained in this section have been taken in the pilot sessions organized by Answare, and provided to ITI through JSON messages.

The analysis carried out consists of making a model of Markov chains of transference between the emotional states of the user at the beginning and at the end of the sessions. Using the historical data, the predictive capacity is evaluated according to the grouping of the data, being the three degrees of grouping considered: (1) the global; (2) grouped by severity of the illness; and, finally (3) the individual history of each user.

The transfer matrices are represented in figures 1, 2 and 3, for global cases, group cases and finally, an individual example case, respectively.


Figure 1: Probabilities of transference between emotional states of all patients and sessions in the case of use of Social Integration according to ITI analysis.



Figure 2: Probabilities of transference between emotional states by severity classification in the case of use of Social Integration according to ITI analysis.

 A Naive Bayes Classifier is implemented by selecting the final emotion that maximizes the probability of transition and evaluating its effectiveness in each grouping case using the Leaving One Out method, which consists of leaving a sample out to evaluate its prediction and calculating the probabilities with the rest, iteratively until all the samples have been exhausted. The effectiveness of the global model is 47.5%, the group is represented in table 1, and the individual has an average precision weighted by populations of 57.6%, but the distribution of values is wide, with a minimum value of 15%, maximum at 100%, mean of 66.67% and standard deviation of 27.7%.


Figure 3: Probabilities of transference between emotional states of an individual in the case of use of Social Integration according to ITI analysis.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ligero** | **Medio-bajo** | **Medio-Alto** | **Grave** | **Media ponderada** |
| 60.8% | 59.0% |  28.3% |  78.9% | 58.4% |

Table 1: Precisions of the model according to the severity of the disorder, and weighted average according to the size of the sample by group.

It can be observed that the individual grouping is too variable, so grouping can improve the results. Therefore this simple example demonstrates the effectiveness of the proposed model, which can be fine-tuned by adding additional physiological response data to it, or by implementing a more complex model.

### User profile acquisition tool

To obtain user profiles, in the case of social integration use, for recreation of the scenes with the visit to the dentist and the hairdresser for autistic children, we have conducted interviews with specialists and monitors who have given us guidelines for creating the scenes. These recommendations have been taken into account in order to create an experience as real and effective as possible in order to meet the objective of preparing autistic children for these situations.

The users of the simulation have been chosen by the specialists themselves from among the children with the best predisposition to use electronic devices of this type.

Once the training process begins, users are identified by an ID assigned by the supervisor. In addition, certain parameters can be configured by the supervisor to match the simulation to the specific individual. During the execution of the application, the scene is adapted to the user according to the emotional state in which he is.

In our case, the tool for obtaining profiles is the ERL software platform which detects the emotions of the user and the gyroscope of the Virtual Reality glasses with which we detect the stress level of the user. With this information, depending on the type of user detected, we can adapt different parameters of the scene, such as lighting or sound, to reduce stress.

With high stress users the scene lighting becomes warmer, relaxing music plays and we lower the sound intensity of medical devices. However, with users with an excited emotional level, the scene shows a cooler illumination and some sounds are intensified in order to lower those levels of excitement of the individual.

The objective of the simulation is to provide a training platform that will help the autistic child face a real visit to the dentist. That is why it is important to have stable levels of stress and emotion so that children can follow the instructions of the simulation and that the virtual training has good results.