WORKINGAGE

Smart Working Environments for All Ages

D6.1 Ontology

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement N. 826232

WP6 – Data Analysis

D6.1 – Ontology

Project number:	826232
Project acronym:	WorkingAge
Project title:	Smart Working Environments for All Ages
Author(s):	UCAM & POLIMI
Partners contributed:	UCAM, POLIMI, BS, INTRAS, RWTH, MUTUA
GA Delivery date:	M21 (Nov 2020)
Document date:	30/11/2020
Version:	1
Revision:	7
Deliverable type:	Report
Remarks:	
Status:	○ PU (Public) ○ PP Restricted to other programme participants (including the Commission Services) ○ Restricted to a group specified by the consortium (including the Commission Services) (please specify the group) ⦿ Confidential, only for members of the consortium (including the Commission Services

Document Revision Log

VERSION	REVISION	DATE	DESCRIPTION	AUTHOR
1	0	04/05/2020	First incomplete version of Ontology	T. Xu (UCAM), R. Tedesco (POLIMI), V. Rick (RWTH), H. Gunes (UCAM)
1	1	13/08/2020	First version of Ontology	T. Xu (UCAM), R. Tedesco (POLIMI), A. G. Aberturas (INTRAS)
1	2	05/09/2020	Update based on review and feedback from MUTUA	T. Xu (UCAM), R. Tedesco (POLIMI
1	3	16/11/20	First complete draft of report	H. Gunes (UCAM)
1	4	20/11/20	Feedback and new content incorporated	H. Gunes (UCAM), V. Roca (BS), A. G. Aberturas (INTRAS)
1	5	26/11/20	Second round of feedback incorporated	H. Gunes (UCAM), V. Roca (BS), A. G. Aberturas (INTRAS)
1	6	30/11/20	Final set of feedback incorporated	H. Gunes (UCAM), V. Roca (BS), A. G. Aberturas (INTRAS), M. van Gasteren (ITCL)
1	7	30/11/20	Final review and formatting	M. van Gasteren (ITCL)

Executive Summary

This Deliverable gives an overview of the WorkingAge Ontology developed in Task 6.1.

Section 1 provides a generic introduction including the motivations and the ontology concept, together with the specific languages employed for the development. Section 2 specifically focuses on the WorkingAge Ontology in terms of its elements of worker, profile, advice, task, sensors and smart goals and provides visual depictions of these for clarity. Section 3 goes into technical details of the implementation using OWL and Protégé and describes how reasoning is achieved. Section 5 describes the data sources in terms of characterisation and measurements. Section 6 summarises the conclusions and Section 7 provides possible future work.

Table of Contents

List of Figures

List of Tables

1 Background and Introduction

1.1 Motivation and Introduction

The WAOW Tool provides advices to the users by means of a Decision Making System (DSS) based on logic rules. Such rules predicate on the content of a Knowledge Base (KB) defined as an OWL[1] Ontology. The WA Ontology defines eight main aspects:

- **Worker** is the "core" of the WA Ontology, modelling workers' personal information. Besides, Worker is connected to other concepts shown below.
- **Profile** contains the personal information of the worker.
- **Task** defines the characteristics of the job performed by a worker.
- **Sensor** describes information we extract from sensors we'll deploy.
- **Smart Goal** describes the aims that the workers are suggested to achieve.
- **Goal State** is closely connected to Smart Goal, which shows the worker's progress towards the personalized Smart Goal.
- **Advice** defines the suggestions that can be provided to workers, to improve their working and living condition.
- **Feedback** is provided by workers, to understand their attitude towards the advice and WA tool.
-

We decided to adopt such a model-driven approach, instead of more common data-driven methods, for two main reasons: lack of data for training a good model, and avail-ability of good experts who provided their experience for defining both the WA Ontology and the logic rules.

In the following, we briefly introduce the “building blocks” of the OWL language and describe the content of the WA Ontology.

1.2 The OWL language

An ontology is a formal definition of concepts belonging to a domain; OWL provides a formalism based on Description Logic, where concept is described as:

- **Classes:** a concept is represented as a Class if it is a set; for example, the concept Worker can be represented as a set, since it will contain the set of workers followed by the WA Tool.
- **Individuals:** a concept is represented as an Individual if it is an element of a set; for example, John is an individual as it could be an element of the Class Worker.
- **Attributes:** a concept is represented as an Attribute if it can be seen as a simple attribute of Individuals belonging to a Class; for example, adopting a theory of emotion based on the pair arousal/valence, an Individual belonging to the Class ContinuousEmotion could exhibit two Attributes: arousal and valence.

[1]The Web Ontology Language (OWL) is a W3C specification and is part of the W3C’s Semantic Web technology stack; see: https://www.w3.org/OWL/.

- **Data Type:** this is a predefined collection of types that will characterize any Attribute and define the legal values; for example, arousal and valence could be of Data Type: Integer from 1 to 9.

Being a restriction of RDF[2], the OWL language is based on *triples* and stored as XML files. For the sake of simplicity, however, the following Classes and Individuals will be represented as nodes of a graph (rectangles will represent Classes while ovals will identify Individuals). The arcs of such graph are called *Relationships*. The labels we put on the arcs represent the semantics of the Relationship. In the following we'll make use of some "predefined" Relationships (which carry OWL-defined semantics), such as:

- **instanceOf:** connects an Individual to the Class it belongs to; for example, John instanceOf Worker
- **isA:** connects a Class to its super-class; for example, BodySensor isA Sensor (in other words, BodySensor is a subset of the set Sensor)

Apart from these relationships, new relationships can be introduced, where the semantic is basically represented by the label used to name them. For example, Worker isWorkingOn Task.

In the following, we show Classes, Attributes, and Relationships, which provide the WA Ontology "schema". Such representation will be stored as an XML file. Whenever new data is added to the WA Ontology -a new sensor measure, a new worker, etc.- one or more Individuals will be added and stored in a specialised database (DB) (we are currently testing Apache Jena Fuseki[3].

Finally, note that the WA Ontology permits to retrieve two different views on data:

- **State:** it represents the current state of some Concept; for example, the current emotive state of a given worker.
- **History:** it represents the past state of some Concept; for example, the list of emotive states of a given worker.

Both views are useful and will be exploited by the rules in charge of generating advice. In practice, we added the notion of *timestamp* to all Classes whose Individuals will represent a time series of data (e.g. sensor measurements).

1.3 Graphical Language

The OWL ontology will be described by means of a simple graphical language. Components:

- Rectangles: used for classes; the label into the rectangle is the class' ID (must be unique).

[2] The Resource Description Framework (RDF) is the W3C's standard model for data interchange on the Web; see: https://www.w3.org/RDF/.
[3] See: https://jena.apache.org/documentation/fuseki2/

- Ovals: used for individuals; the table into the oval is the individual's ID (must be unique).
- Pins: used for attributes (datatype properties, in OWL wording)
  - The attribute name is specified with a label.
  - The default arity of the attribute is 1 (one and only one attribute with that label must be present).
  - Attributes that can appear multiple times are marked with a number n; this means that exactly n attributes with the same label must be present).
  - There are two types of pin, white and black.
    - White pin: the attribute with that label must present.
    - Black pin: at least one of the attributes within the same class must present each time.
- Arcs: used for relationships (object properties, in OWL wording)
  - Arcs connect a Domain class to a Range class.
  - Arcs are named by means of IDs (i.e., labels that must be unique)
  - Relationships can belong to the following typologies:
    - 1-to-1: the default typology
    - 1-to-N: the party that can appear multiple times is marker as "0..N" (if it is optional) o "1..N" if at least one must be present
    - N-to-N: both parties are marked as "0..N" (if optional) o "1..N" if at least one must be present.
  - Some relationship typologies are defined by the language:
    - isA: the subclass object property. The sibling subclasses are disjoint and exhaustive (i.e., they constitute a partition of their superclass).
    - InstanceOf: connects an individual to the corresponding class.

2 **The WorkingAge Ontology**

For the sake of simplicity, the WA Ontology is divided into six parts; in the following we provide a brief description for all of them.

2.1 **Worker**

The Class Worker, depicted in Figure 1, is defined with some basic Attributes, like UserID and AddOnDate. Note that UserID is used as a key attribute that is identical for each worker on the system.

Worker is connected to Profile, Task, Sensor, Advice, Feedback and SmartGoal, which represents the information flow at the run time, about the workers.

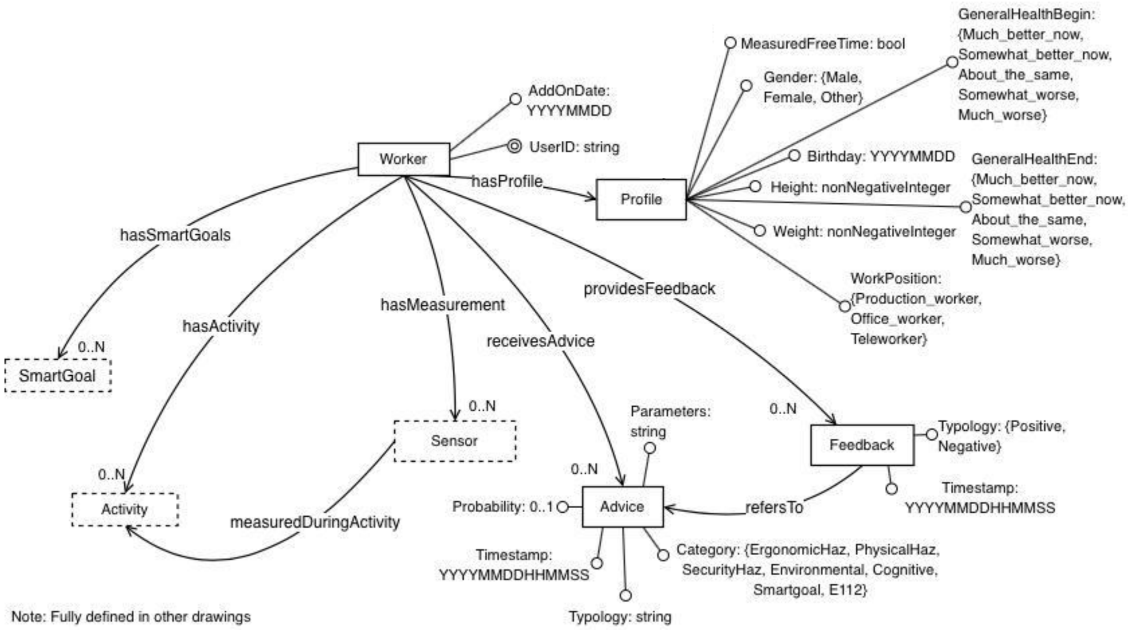

Figure 1. Worker, with related Classes.

2.2 **Profile**

The Class Profile, depicted in Figure 1, contains the static personal information of the worker, such as Gender, Birthday, HeightWeight, WorkPosition, GeneralHealthBegin, GeneralHealthEnd, and MeasuredFreeTime . The Health-related attributes provide the general description of the worker's health condition before and after participating the WorkingAge project.

2.3 **Advice**

The Class Advice and Feedback are depicted in Figure 1. Advice has five Attributes: Timestamp, Typology, Category, Parameters and Probability; Typology contains a label indicating a suggested action, for example "Change_pose"; Category contains a label showing under what circumstances the advice is needed (e.g. "ErgonomicHaz": when Ergonomic Hazard is detected, advice will be provided); Parameters contains an optional comma-separated list of parameters (e.g., the current heartbeat "150").

Notice that the Typology label is not meant to be shown to the worker: once defined the list of possible typology labels, a localized conversion file will permit to transform the label (with all the parameters, if any) into a human-readable message; for example, starting from "Call_E112" and "150", if the localized conversion file contains a row formatted as

"Call_E112 Cardiac anomaly detected; heartbeat: #1. The system is going to call the E112 service"

the following sentence could be generated:
"Cardiac anomaly detected; heartbeat: 150. The system is going to call the E112 service".

Feedback has two Attributes: Timestamp and Typology; Typology provides information about whether the Feedback towards the advice is positive or negative.

Worker receives Advice from WA tool, while the Worker can also provide Feedback to WA tool. According to the Feedback, it is possible to adjust the Advice to better serve Worker.

2.4 **Task**

This Class describes basic pieces of information we need to collect about a job (see Figure 2).

So far, we described three job typologies: Office, Assembly, and Teleworking, as the WA project previews such use cases. The FreeTimeAtHome Class refers to off-work home activities, as the WA project aims at considering the whole daily life of the worker.

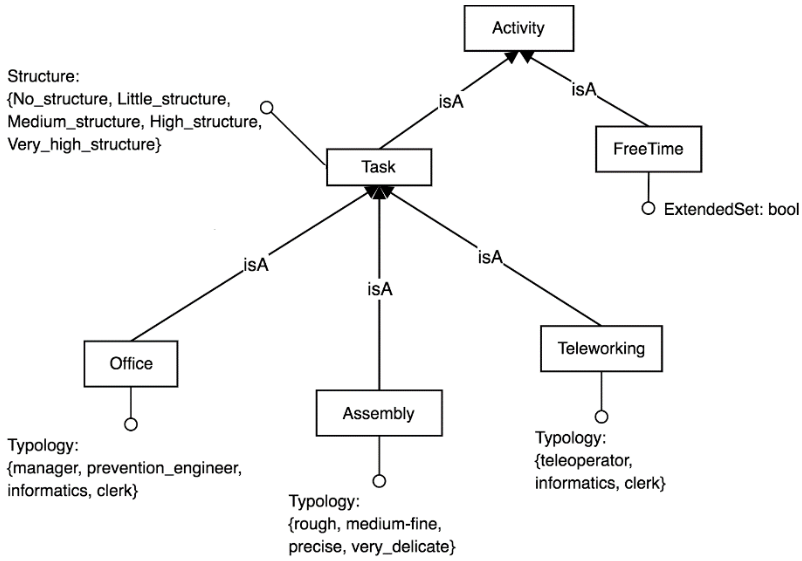

Figure 2 Task, with related Classes.

2.5 **Sensor**

As Figure 3 shows, this Class is the root of the most complex taxonomy in the WA Ontology. Here, we describe the characteristics of information we can derive from sensor measurements.

Sensors are divided into three main groups:

- Sensors that measure the environment
- Sensors that measure the worker's body
- Other sensors

The Class EnvironmentalSensor is the root of sensors that collect data from the environment. The subclass ThermoHygrometric measures the temperature, humidity and CO2 concentration discomfort. The subclass EnvMicrophone measures the environmental and human noise level. Light measures the illumination level.

The Class BodySensor is the root of wearable sensors, which collect a vast variety of data, such as wearable bio-metric devices (for example, ECG and GSR), cameras (for example, body posture, facial expression and eye movements), and microphones (voice recording), etc. From such sensors we can derive several information. Below is the full list of the information that can be derived from each body sensor.

- Neurophysiological: Stress Level, Emotional State, Mental Workload, Heart Rate
- FacialExpress (Facial Expression): Arousal, Valence
- EyeMove (Eye Movement): Mental Strain Level
- BodyPosture: Sitting Risk (RULA), Standing Risk (REBA)

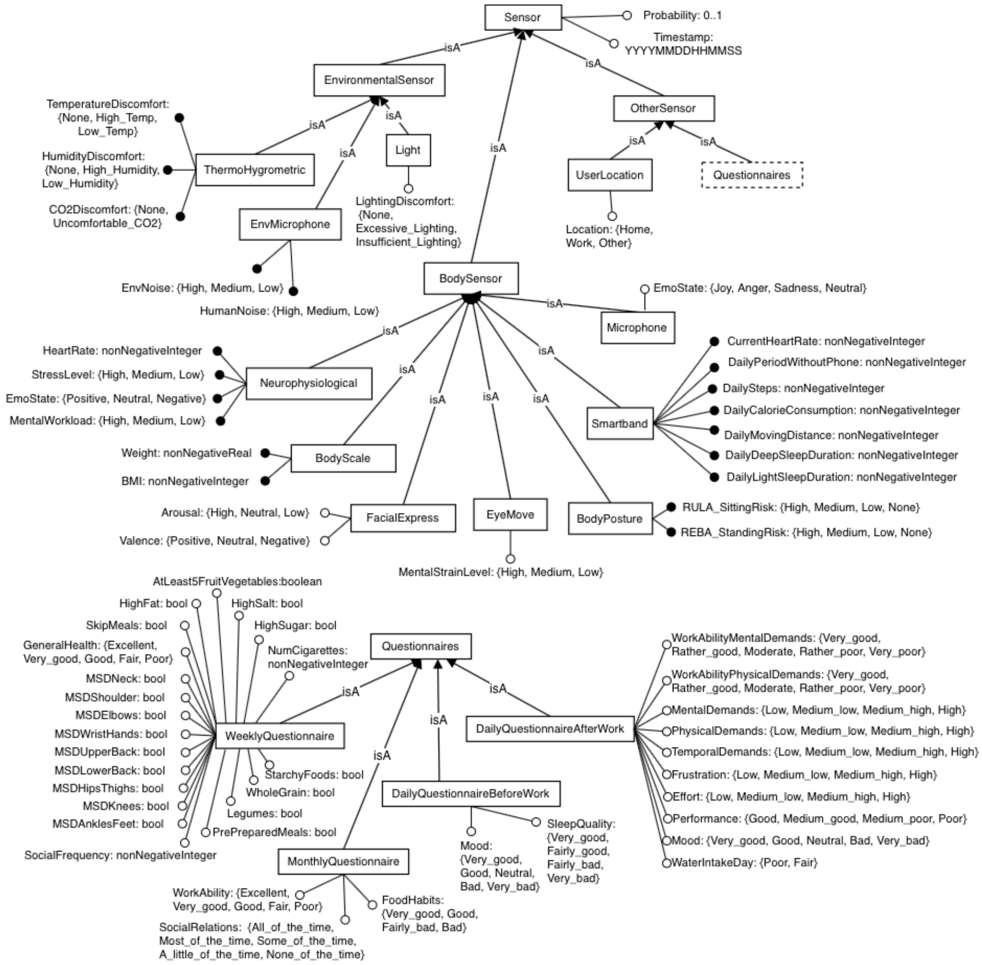

Figure 3 Sensor, with related Classes.

Smartband: Current Heart Rate , Period Without Phone before sleeping[4], Steps in a day, Calorie Consumption in a day, Moving Distance in a day, Deep Sleep Duration in a day, Light Sleep Duration in a day

- Microphone: Emotional State
- BodyScale: Weight, BMI

The Class OtherSensor is currently used to represent collected information that does not belong to the environmental and body sensor, such as UserLocation and Questionnaires. The Class UserLocation defines the location information coming from GPS and/or indoor positioning systems. The Class Questionnaires will be regularly administered to workers, to understand their current states, as a special case of "sensors". Currently there are four categories of the Questionnaires which are listed below.

[4] In our current implementation, this one and the other "daily" attributes are updates in the morning, so they refer to the day before the one indicated by the timestamp.

- DailyQuestionnaireBeforeWork (Daily Questionnaire Before Work): Mood, Sleep Quality
- DailyQuestionnaireAfterWork (Daily Questionnaire After Work): Work Ability Mental Demands, Work Ability Physical Demands, Mental Demands, Physical Demands, Temporal Demands, Frustration, Effort, Performance, Mood, Water Intake per Day
- WeeklyQuestionnaire (Weekly Questionnaire): General Health, MSD[5] neck, MSD Shoulder, MSD Elbows, MSD Wrist Hands, MSD Upper Back, MSD Lower Back, MSD Hips Thighs, MSD Knees, MSD Ankles Feet, Social Frequency, Skip Meals, High Fat, High Salt, High Sugar, Whole Grains, Legumes, Pre-Prepared Meals, Starchy Foods, At Least 5 Fruit Vegetables, Num. Cigarettes
- MonthlyQuestionnaire (Monthly Questionnaire): Work Ability, Food Habits, Social Relations

Note the Timestamp Attribute, which permits to extract a history of measures associated to a give worker, which is another part of the current *state* of a worker. The Attribute Probability indicates the confidence level of the sensor outputs.

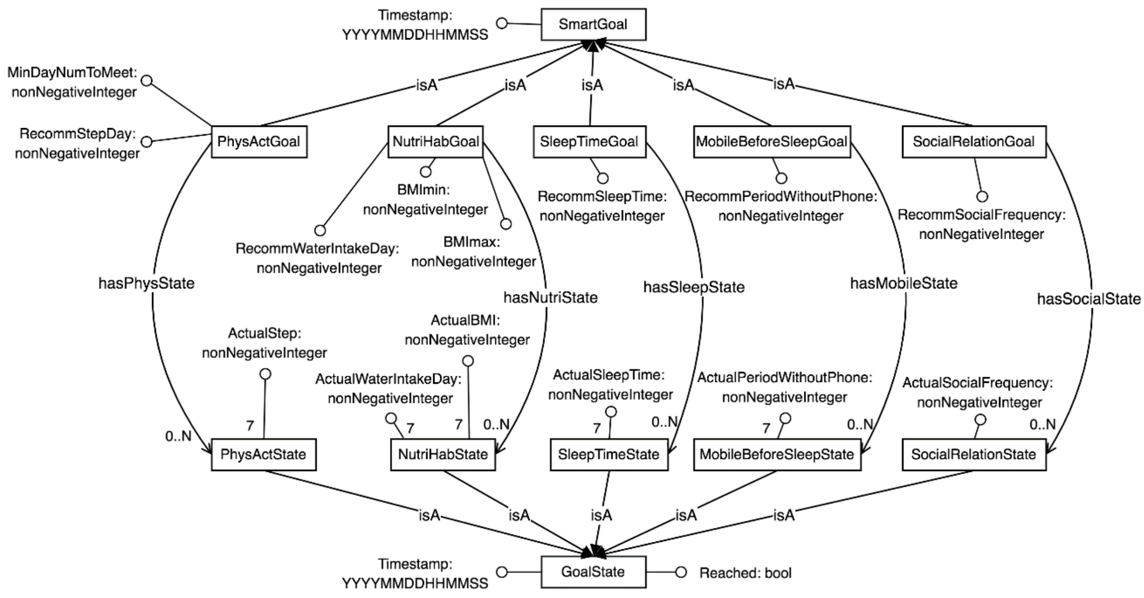

Figure 4 Smart Goal, with related Classes

2.6 **Smart Goal and Goal State**

According to the worker's situation, SmartGoal will be set for individuals. These personalized goals can help worker keep healthy lifestyle and improve their

[5] Musculoskeletal Disorder (MSD) risk.

working and living condition. The five aspects of the Smart Goal are shown below.

- PhysActGoal (Physical Activity Goal): Recommended Steps per Day, Min Daily Steps to Meet
- NutriHabGoal (Nutrition Habits Goal): Recommended Water Intake per Day, BMI min, BMI max
- SleepTimeGoal (Sleep Time Goal): Recommended Sleeping Time
- MobileBeforeSleepGoal (Mobile phone Usage Before Sleep Goal): Recommended Period Without Phone
- SocialRelationGoal (Social Relations Goal): Recommended Social Frequency
- The Class GoalState is to show the worker's progress towards the set Smart Goal. Each subclass of Smart Goal has a corresponding Goal State class, as shown in Figure 4.

The Smart Goals will be assessed along the experimental phases of the project for each user. The relationship between the sensors and the Smart Goals is shown below.

- PhysActState (Physical Activity State): it will be assessed considering the high-level information provided by the Xiaomi Mi Band 4, the smartband. It will be assessed considering the number of steps.
- NutriHabState (Nutrition Habits State): it will be assessed considering the high-level information (i.e. Actual BMI) provided by the Xiaomi Mi Scale, the smart balance. The questionnaire will provide the information on the Actual Water Intake per Day.
- SleepTimeState (Sleeping Time State): it will be assessed considering the high-level information provided by the Xiaomi Mi Band 4, the smartband. It will be assessed considering the sleep duration.
- MobileBeforeSleepState (Mobile phone Usage Before Sleep State): it will be assessed considering the Android information related to the screen usage.
- SocialRelationState (Social Relations State): it will be assessed considering the questionnaires provided through the WA mobile app.

2.7 **Adding individuals**

So far, we described the "schema" of the WA Ontology. But, at run time, the WA Ontology should be populated by Individuals (new workers, new sensor measurements, new tasks, new social environments.)

Figure 5 shows a possible example, where a new worker is added to the WA Ontology. Then, two measures of emotion are added, starting from voice recordings; in practice, two new Individuals of Microphone are added to the WA Ontology, each of them named with an UUID. Note the arcs connecting the worker to its measurements. Moreover, note that the two measurements are marked with a timestamp; thus, we can reconstruct the history of emotions expressed by the worker so far, as well as her/his "current emotive state" (i.e., the last measured emotion).

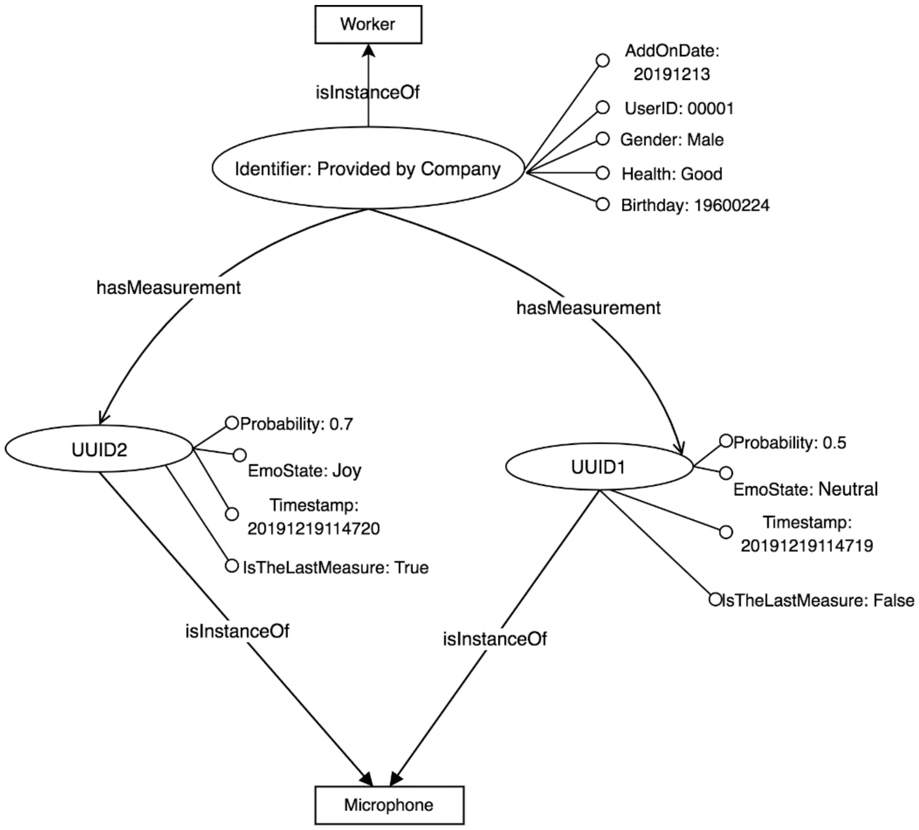

Figure 5 Two measurements of emotion, based on voice recordings.

In order to improve efficiency, Individuals are directly inserted into the ProbLog engine, which oversees performing reasoning.

3 **OWL and Protégé**

Implementation details about the WA Ontology implementation using OWL by means of the Protégé editor.

General axioms:

Each class in the taxonomy is defined as a "Disjoint Union Of" its offspring classes (i.e., sibling classes constitute a partition of their parent class).

Attribute axioms:

Attribute userID of class Worker is defined as a “Target for Key”: basically, it is defined as a unique key.

4 **Reasoning**

As mentioned at the beginning of this report, a set of rules will be used to perform reasoning and derive the advice and the suggestions that will be provided to the workers.

From the beginning we chose to describe our Knowledge Base (KB) as an Ontology, and an OWL-based Ontology. OWL is the *de-facto* standard for describing Ontologies, and very useful tools exist that support authoring of OWL-bases Ontologies. We chose to use Protégé, which provides a powerful and extensible environment. As OWL is a well-known and well-supported language, exporting our Ontology to other formats would be easy.

Powerful libraries exist that enable the manipulation of the Ontology using several programming languages. Initially we considered using SWRL, which is part of the OWL specification and can directly work on the OWL Ontology. Unfortunately, SWRL is not well suited for working with probabilistic information. Some of the measurements extracted from the sensors are probabilistic; as an example, the emotional state of Microphone is calculated by a statistical classifier that provides a probability associated with the emotion class it predicts. Due to this, we switched to ProbLog, which can deal with such probabilistic information and can perform probabilistic reasoning.

5 Data sources

5.1 Characterisation

The following tables, from Table 1 to Table 1, which derive from D2.5, describe data sources in terms of measurements, sensors, WA partner responsible for that data collection and management, etc.

5.2 Measurements and the Ontology

In this section, tables from Table 6 to Table 10 describe how each Ontology attribute should be calculated, by means of the measurements described above.

Each Ontology[6] attribute is associated to a measurement (i.e., an information source), which is calculated starting from raw-data provided by a sensor (i.e., a data source). Table 1 shows the relationship between measures and sensors.

In the following tables, for each attribute belonging to the Ontology, we provide how its value is calculated[7].

Table 1 Measurements and related sensors

Measurement	**Sensor**
Heart activity	ECG (Empatica wristband)
Galvanic Skin Response	GSR (Empatica wristband)
Brain activity	EEG headband
Facial expression	Camera #1
Voice analysis	Unidirectional, noise-cancelling mic
Eye blink detection	Headband or Camera #1 (*)
Eye movement, pupil diameter	Eye tracker
Body posture	Camera #2
Gesture recognition (**)	Camera #1
User location	Via smartphone

[6] In synch with the content of the OWL file "workingage-v4.2.4.owl" and the PNG images.

[7] For questionnaires, see "T3.1 Questionnaires Ontology INTRAS.xlsx"

Sleep duration and quality	Smartband
Step meter	Smartband
Heart rate	Smartband
Weight	Body scale
BMI	Body scale
Questionnaires (***)	Questionnaires on WA App
- Daily (before work)	
- Daily (after work)	
- Weekly	
- Monthly	
- Static (profile)	
Noise	Omnidirectional microphone
- Environmental	
- Human	
Lux, thermo-hygrometric, CO_2	Environment Condition sensor
(*) To be decided according to the In-Lab tests results.	
(**) Only for controlling the WA App UI. Removed from following tables.	
(***) Multiple measurements on work environment and working conditions; cognitive, emotional and social aspects; nutrition, sleep, exercise; dynamic and kinesthetics characteristics of the task, etc.; about day-to-day social interaction; initial interview.	

Table 2 Subjective vs objective; subject vs environmental measurements

Measurements	**Objective / Subjective**	**Subject / Environment**
Heart activity	O	S
Galvanic Skin Response	O	S
Brain activity	O	S
Facial expression	O	S
Voice analysis	O	S
Eye blink detection	O	S
Eye movement, pupil diameter	O	S
Body posture	O	S
User location	O	S

Sleep duration and quality	O	S
Step meter	O	S
Heart rate	O	S
Weight	O	S
BMI	O	S
Questionnaires	S	S
Noise	O	E
Lux, thermo-hygrometric, CO_2	O	E

Table 3 Measurements and use cases (X: only workers with home extension list)

	Use cases			
Measurements	**Office**	**Teleworking**	**Manufacturing**	**Daily life**
Heart activity	✓	✓	✓	X
Galvanic Skin Response	✓	✓	✓	X
Brain activity	✓	✓	✓	X
Facial expression	✓	✓	✓	X
Voice analysis	✓	✓	-	X
Eye blink detection	✓	✓	✓	X
Eye movement, pupil diameter	✓	✓	✓	-
Body posture	✓	✓	✓	X
User location	✓	✓	✓	X
Sleep duration & quality	-	-	-	✓
Step meter	✓	✓	✓	✓
Heart rate	✓	✓	✓	✓
Weight	-	-	-	✓
BMI	-	-	-	✓
Questionnaires	✓	✓	✓	✓
Noise	✓	✓	✓	X
Lux, thermo-hygrometric, CO_2	✓	✓	✓	X

Table 4 Number of subjects involved in measurements. All: all subjects involved in the use case (also equipped with the home standard sensor list); Some: sub-jects who agreed to be equipped with home standard + home extension sensor lists

Measurement	**Per Use case**	**Daily life (all use cases)**
Heart activity	3 subjects	Some, max. 9 subjects

Galvanic Skin Response	3 subjects	Some, max. 9 subjects
Brain activity	2 subjects	Some, max. 6 subjects
Facial expression	All	Some
Voice analysis	All(*)	Some
Eye blink detection	All / 2 subjects(**)	Some / Some, max. 6 subjects(**)
Eye movement, pupil diameter	1 subject	-
Body posture	All	Some
User location	All	Some
Sleep duration and quality	-	All
Step meter	All	All
Heart rate	All	All
Weight	-	All
BMI	-	All
Questionnaires	All	All
Noise	All	Some
Lux, thermo-hygrometric, CO_2	All	Some
(*) Excluding the "Factory" use case.		
(**) In case the headband is used, only 2 subjects will be monitored per use case; in case the camera is used, there is no constraint.		

Table 5 Measure frequency (in terms of the derived High-level information put into the Ontology); WA personnel needed?

Measurements	**Frequency or event**	**Attended/ Unattended**	**Owner**
Heart activity, Galvanic Skin Response, brain activity	8 Hz; 1/min (++)	U	BS
Facial expression	1/min (+)	U	UCAM
Voice analysis	Event: after each utterance	U	POLIMI/AUD
Eye blink detection	1/min	U	BS
Eye movement, pupil diameter	1/min (*)	A	RWTH
Body posture	1/s	U	ITCL
User location	On location	U	TPZ

	change (+++)		
Sleep duration and quality	1/day (***)	U	ITCL
Step meter	1/day (***)	U	ITCL
Heart rate	1/min	U	ITCL
Weight	User specified	U	ITCL
BMI	User specified	U	ITCL
Questionnaires:		U	INTRAS
- Daily (before work)	1/day		
- Daily (after work)	1/day		
- Weekly	1/week		
- Monthly	1/month		
- Static (profile)	once		
Noise	1/min	U	AUD
Lux, thermo-hygrometric, CO_2	Events: sensor out of threshold ranges for at least 30 minutes; sensor back within threshold.	U	ITCL
(++) High-level info generated by sensor fusion, every minute; for the raw heart rate (for the E112 service experiment), max freq is 8 Hz. (+++) The positioning algorithm is sampled 1/min, but the High-level info is generated only when the position is changing. (+) Or more, if needed. (*) Up to 200 Hz, if needed. (***) Sensor sampled every 30 minutes; High-level info sent every day.			

Table 6 Calculating attributes; Worker-related

Class: Attribute	**Measure and calculation**

`Worker:`
`AddOnDate`
`UserID`
	
	WA App login process (the current day)
	WA App login process (provided by the worker)

Profile: Gender BirthDate Height Weight WorkPosition GeneralHealthBegin GeneralHealthEnd MeasuredFreeTime	 Static questionnaire Static questionnaire Static questionnaire Static questionnaire Static questionnaire SF36, static questionnaire, Health_OneYear SF36, static questionnaire, Health_OneYear Static questionnaire
Advice: Probability Timestamp Typology Category Parameters	 Generated by the DSS Generated by the DSS Generated by the DSS Generated by the DSS Generated by the DSS
Feedback: Typology Timestamp	 From the WA App UI for feedback mngt Generated by the WA App: current time

Table 7 Calculating attributes; Task-related

Class:Attribute	**Measure and calculation**
Activity: -	none
Task: Structure	Static questionnaire
FreeTime: ExtendedSet	Static questionnaire
Office: Typology	Static questionnaire
Assembly: Typology	Static questionnaire
Teleworking: Typology	Static questionnaire

Table 8 Calculating attributes; Smart goals

Class:Attribute	**Measure and calculation**

`SmartGoal:` `Timestamp`	Current timestamp
`MobileBeforeSleepGoal:` `RecommPeriodWithoutPhone`	WA App, hardcoded

`NutriHabGoal:`
`BMImax`
`BMImin`
`RecommWaterIntakeDay`
	WA App, hardcoded
	WA App, hardcoded
	WA App configuration; min 1.5l

`PhysActGoal:`
`MinDayNumToMeet`
`RecommStepDay`
	WA App configuration; min 4
	WA App configuration; min 6000

`SleepTimeGoal:` `RecommSleepTime`	WA App configuration; 7.5h
`SocialRelationGoal:` `RecommSocialFrequency`	WA App configuration; min 1
`GoalState:` `Timestamp`	Current timestamp
`MobileBeforeSleepState:` `ActualPeriodWithoutPhone`	7 measures; see: SmartBand:DailyPeriodWithoutPhone

`NutriHabState:`
`ActualBMI`
`ActualWaterIntakeDay`
	7 measures; see: BodyScale:BMI
	7 answers;
	see:
	DailyQuestionnaireAfterWork:WaterIntakeDay

`PhysActState:` `ActualStep`	7 measures; see: Smartband:Steps
`GoalState:` `ActualSleepTime`	7 measure; see: SmartBand:DailyDeepSleepDuration + SmartBand:DailyLightSleepDuration
`SocialRelationState:` `ActualSocialFrequency`	See: WeeklyQuestionnaire:SocialFrequency

Table 9 Calculating attributes; Sensors

Class:Attribute	**Measure and calculation**
Sensor: Probability Timestamp	Probability of the current high-level info Current timestamp
BodyPosture: REBAStandingRisk RULASittingRisk	Camera #2: body posture Camera #2: body posture
BodyScale: BMI Weight	Body scale Body scale
EyeMove: MentalStrainLevel	Eye tracker (eye movement + pupil diam.)
FacialExpress: Arousal Valence	Camera #1: facial expresión Camera #1: facial expression
Microphone: EmoState	Unidirectional, noise-cancelling mic
Neurophysiological: HeartRate EmoState StressLevel MentalWorkload	ECG ECG + GSR EEG + ECG + GSR/Eye blink detection EEG
Smartband: DailyLightSleepDuration DailyDeepSleepDuration DailySteps DailyMovingDistance DailyCalorieConsumption DailyPeriodWithoutPhone CurrentHeartRate	Smartband Smartband Smartband Smartband Smartband Smartband Smartband
EnvMicrophone: HumanNoise EnvNoise	Omnidirectional microphone Omnidirectional microphone

Light: LightingRisk	Illumination
ThermoHygrometric: HumidityDiscomfort CO2Discomfort TemperatureDiscomfort	 Environment Condition sensor Environment Condition sensor Environment Condition sensor
UserLocation: Location	Smartphone

Table 10 Calculating attributes; Questionnaires

Class:Attribute	**Questionnaire name**
DailyQuestionnaireAfterWork: Effort Frustration MentalDemands Mood Performance PhysicalDemands TemporalDemands WaterIntakeDay WorkAbilityMentalDemands WorkAbilityPhysicalDemands	 NASA-TLX NASA-TLX NASA-TLX NASA-TLX NASA-TLX NASA-TLX NASA-TLX Ad-hoc question WAI 2 WAI 2
DailyQuestionnaireBeforeWork: Mood SleepQuality	 NASA-TLX Pitsburg Sleep Quality Index
MonthlyQuestionnaire: FoodHabits WorkAbility SocialRelations	 How Healthy Is Your Diet WAI 1 (WorkAbility) SF36

WeeklyQuestionnaire:	
GeneralHealth	SF36 (Curr_Health)
MSDAnklesFeet	NMQ
MSDElbows	NMQ
MSDHipsThighs	NMQ
MSDKnees	NMQ
MSDLowerBack	NMQ
MSDNeck	NMQ
MSDShoulder	NMQ
MSDUpperBack	NMQ
MSDWristHands	NMQ
SocialFrequency	Ad-hoc question
NumCigarettes	Ad-hoc question
AtLeast5FruitVegetables	Ad-hoc question
HighSalt	Ad-hoc question
HighSugar	Ad-hoc question
StarchyFoods	Ad-hoc question
WholeGrain	Ad-hoc question
Legumes	Ad-hoc question
PrePreparedMeals	Ad-hoc question
SkipMeals	Ad-hoc question
HighFat	Ad-hoc question

6 Conclusions

The WA Ontology presented in this document provides the WAOW Tool with a common vocabulary all the WA partners should conform to. By doing so, whenever different partners need to share information, the WA Ontology eliminates incompatibilities. As it is defined in a formal manner, the WA Ontology is also a schema that can be used as a reference for data manipulation and therefore it is a reference for the DSS. As the WA ontology was implemented using the standard OWL language, it provides means for checking the correctness of messages exchanged among the WAOW Tool components.

Defining the WA Ontology, and implementing it with a formal language like OWL, has had other benefits during the initial steps of the WA project, i.e., it forced us to reason and precisely define the knowledge we needed to represent, in order to reach the goals of the WA project.

The WA Ontology has been shaped by the three use cases of the WA project, but most of it is generic enough to be reused in other contexts. The parts that are related to the three use cases are easily recognized and therefore can be easily edited to adapt them to new scenarios using the well-known and commonly used language OWL.

7 Future Work

For our first prototype, the WA Ontology is not employed for checking messages. We plan to add this to future versions, in order to improve the robustness of the WA communication infrastructure.

8 References

[1] D. Fierens, G. Van den Broeck, J. Renkens, D. Shterionov, I. Thon B. Gutmann, G. Janssens, and L. De Raedt. Inference and learning in probabilistic logic pro-grams using weighted boolean formulas. Theory and Practice of Logic Programming, 15(3):358–401, 2015.