Project ref no	AAL-2012-5-195
Project acronym	DIET4Elders
Project full title	Dynamic nutrItion bEhaviour awareness sysTem FOR the Elders
Dissemination level	Public
Date of delivery	12/15/2014
Deliverable number	D4.1
Deliverable name	First version of the DIET4Elders self-feeding support services
Type	Report and prototype (R/P)
Status	Final
WP contributing to the deliverable	WP4
Main contributors	TUC
Other contributors	<KCL: ISOIN: COESCO: TUNSTALL>
Author(s)	< TUC (Ioan Salomie; Tudor Cioara; Ionut Anghel), KCL (Adel Taweel; Simon Miles; Lina Barakat; Dianne Reidlinger), ISOIN (Elisa Jimeno, Alejandro Garcia Marchena, Victor Sanchez)>
Keywords	< reliability-aware marketplace, reliability dimensions, food service model, bio-inspired heuristics, support services development >
Abstract (for dissemination)	In this deliverable we present achievements obtained as a result of the scientific work carried out in the WP4. The first achievement is the DIET4Elders reliability-aware marketplace for food delivery services. The marketplace is based on rich semantic models for capturing the nutritional aspects and various other food-related characteristics as well as reliability dimensions of the registered services. The second achievement is the definition of hybrid bio-inspired techniques which are used to enact the older adult diet aware dynamic discovery and composition of food services. We have defined two such heuristics Hybrid Invasive Weed and Particle Swarm Optimization. The third achievement is the design and implementation of the four DIET4Elders support services: Nutrition Monitoring Service, Nutrition Assessment Service, Nutrition Problem Identification Service and Nutrition Intervention Service.

Dynamic nutrition bEhaviour awareness system FOR the Elders

AAL-2012-5-195

Deliverable   **D4.1**
First version of the DIET4Elders self-feeding support services

Public

VERSION HISTORY

Version	Edited by	Date	Description
0.1	TUC	03/11/2014	Table of Contents
0.9	TUC, KCL, ISOIN	18/11/2014	Deliverable version for peer review
0.95	ISOIN	03/12/2014	Reviewed version
0.95	KCL	09/12/2014	Reviewed version
1.0	TUC	15/12/2014	Final Version

Table of Contents

1. Introduction

The urgency of the increasing size of the aging population is placing a tremendous burden on health and social services in many of the European countries. Enabling older adults to live independently at home is seen one of the promising ways forward, perceived best for both the individuals as well as their carers. One of the critical issues affecting elderlies' ability to live independently at home can be the un-healthy self-feeding behavior which may lead to malnutrition [1].

The DIET4Elders (Dynamic nutrItion bEhaviour awareness sysTem FOR the Elders) project aims to create a system that will provide a set of support services to enable older adults to self-manage their nutritional intake, thus preventing unhealthy self-feeding habits and helping them to follow a healthy diet.

Following a healthy diet plan that meets an older adult's nutritional needs, while considering their health profile and food preferences, is a challenging task. The older adults' ability to cook and prepare food may be affected by various ageing-related diseases, thus they have to rely on external food providers in order to maintain a well-balanced diet. To address the above presented problem, the DIET4Elders solution is to define and develop an electronic marketplace supporting the publication, description, discovery, and composition of food services. The marketplace is based on rich semantic food service model (i.e. Food Service Description Model) which will capture the nutritional aspects and various other food-related characteristics (such as ingredients, cuisine, etc.), thus enabling diet-aware and preference-based selection of food for older adults. Also a reliability model for the food services registered within the marketplace is defined and used to capture a relevant number of reliability dimensions such as: computational reliability, product reliability (i.e. food quality aspects, nutrition aspects, delivery aspects) and fault tolerance.

Based on the developed food services marketplace, the selection and optimal combination of food services satisfying complex criteria such as the older adult recommended diet or food preferences is an NP-hard (Non-deterministic Polynomial-time hard) problem which cannot be solved in reasonable time using conventional techniques.

The DIET4Elders solution is to model the problem as a combinatorial optimization problem and solve it using hybrid bio-inspired techniques that should combine the strength elements of different bio-inspired meta-heuristics. In this deliverable we define and use two bio-inspired meta-heuristics for solving the diet aware food ordering problem: Hybrid Invasive Weed Heuristic and Particle Swarm Optimization (PSO). For defining both meta-heuristics the following steps are followed: (1) the optimization problem is formally represented and the search space defined, (2) the concepts of the optimization problem are mapped to the meta-heuristic concepts and appropriate fitness functions are defined, and (3) the generic meta-heuristic's algorithm is adapted and enhanced to deal with the specificity of the diet aware food ordering problem.

Finally this deliverable addresses the problem of DIET4Elders support services design and implementation. From ICT and ambient assisted living we adopted the MAPE (Monitoring, Analysis, Planning and Execution) architecture of designing context aware adaptable systems. A context aware adaptive system must understand the context in which it evolves (i.e. understand the context changes) and automatically decide on the actions that need to be executed to adapt its execution to the context changes. Form the older adult nutrition perspective we have used the knowledge and expertise regarding the nutrition care process. By combining them we have designed and implemented four support services for older adults:

- Nutrition Monitoring Service whose objective is to acquire data regarding the older adult nutrition, profile and food preferences.
- Nutrition Assessment Service whose objective is to evaluate the older adult nutrition (based on the monitored data) with the goal of identifying unhealthy older adult behaviours.
- Nutrition Problem Identification Service whose objective is the early identification and assessment of potential nutrition related problems and associated symptoms based on the monitored data.
- Nutrition Intervention Service whose objective is to provide for the older adults nutrition education, dietary recommendations and diet aware food ordering.

The rest of this deliverable is organized as follows:

- Section 2 presents the reliability-aware marketplace for food delivery services;
- Section 3 details the hybrid bio-inspired techniques defined and used to enact the diet aware food services dynamic discovery and composition;
- Section 4 presents the DIET4Elders support services design and implementation;
- Section 5 concludes the deliverable.

2. Reliability-aware Marketplace for Food Delivery Services

Following a healthy diet plan that meets an older adult's nutritional needs, while accommodating their medical conditions and food preferences, is essential for addressing the malnutrition problem. Whilst those living on their own may wish (and are encouraged) to remain independent, their ability to cook and prepare food may be affected by various aging-related diseases, making it difficult for them to maintain a well-balanced diet. Utilizing external food providers thus provides an opportunity for such older adults to obtain the quality, nutritious meals they are no longer capable of cooking at home. Moreover, meal delivery services can also bring other benefits for older adults including cost-efficiency and social interaction.

Such reliance on external food providers, however, also imposes challenges. Although a range of food services supporting older adults in achieving a healthy diet may be available, potential clients (older adults, their carers, or software applications acting on their behalf) might not be aware of these services, thus demanding a means to increase their visibility. Additionally, such food services need to be sufficiently described in order to enable an appropriate selection (e.g. to ensure the meals selected are both appealing and comply with specific dietary criteria). Finally, the reliability of such services is an important concern, given the external and possibly commercially driven nature of providers.

In response, we propose an electronic marketplace supporting the publication, description, discovery, and monitoring of the various food services available for older adults. Specifically, it provides a place where food providers (e.g. commercial suppliers or charity organizations) can register their offered food delivery services for older adults, and potential clients can search for suitable services. The services registered within the marketplace will conform to a rich semantic description specification, capturing necessary nutritional aspects and various other food-related characteristics (such as ingredients, cuisine, etc.), thus enabling diet-aware and preference-based selection of food services for older adults. Furthermore, through appropriate performance monitoring and prediction mechanisms, the reliability of the services registered is assessed with respect to a number of relevant dimensions, ultimately ensuring a more trustworthy environment.

2.1. Marketplace Architecture

We follow a service-oriented paradigm to provide a suitable architectural design for the marketplace, with two main components being distinguished: service broker and service reliability manager (see Figure 1).

The service broker allows providers to publish information about the food services they support for older adults through suitable registration interfaces, and organizes such information within a dedicated registry to facilitate its discovery by clients. Core to the registry is a semantic meta-model for food services, facilitating uniform and comprehensive description of the food providers and their offered food delivery services with respect to various nutritional and food aspects. In particular, this description is conducted according to the *Food Provider Ontology* that is developed for this purpose (see Section 2.2 for details on this ontology).

Clients can contact the service broker to locate appropriate food services and obtain their access details. In the context of the DIET4Elders project, the registry is searched by a diet-aware food service selection and composition engine, which selects and integrates food services into optimal combinations for older adults.

The service reliability manager is concerned with monitoring and assessing a number of reliability dimensions for food services, guided by the *Reliability Monitoring Ontology* that is developed for this purpose (see Section 2.3.2 for details on this ontology). Specifically, the reliability manager maintains a repository of past performance data concerning the registered services, collected either through the ambient assisted living infrastructure or through appropriate older adult feedback interfaces. This observation history is then used to predict the characteristics of food services (with respect to the reliability dimensions of interest) via applying appropriate learning mechanisms. The learned characteristics are correspondingly reflected in the registry, to facilitate more accurate and reliable service descriptions.

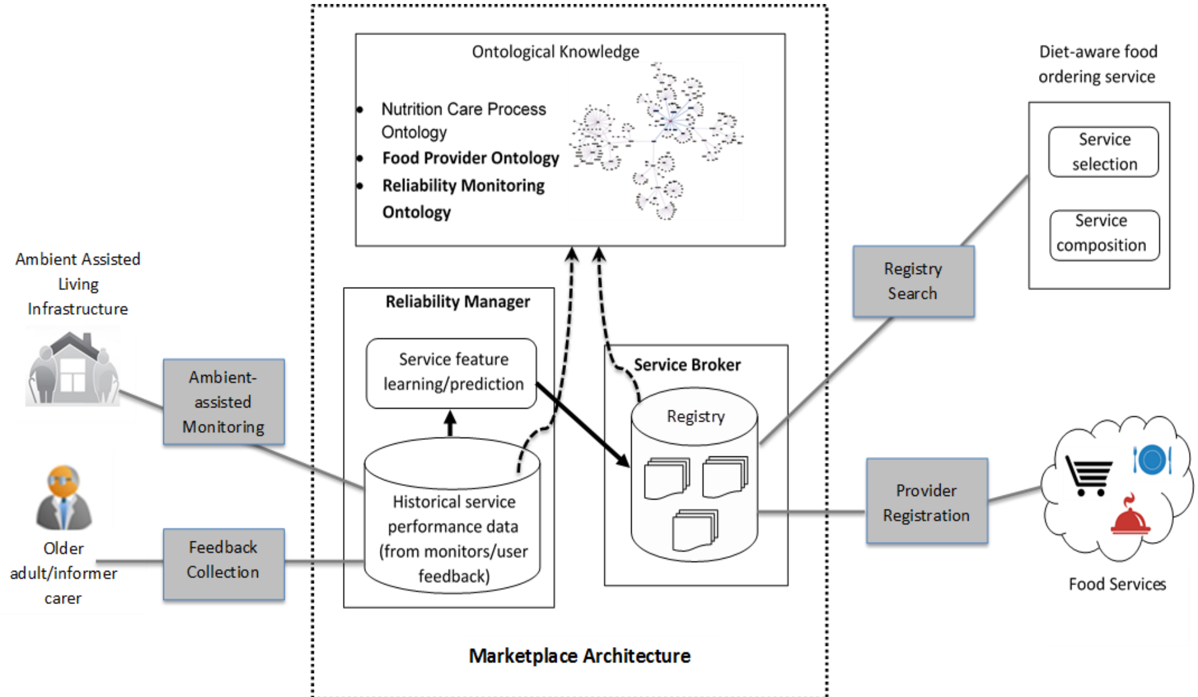

Figure 1. Food Service Marketplace Architecture

2.2. Food Service Description Model

As stated earlier, an important goal of the marketplace is facilitating a standardized, nutritionally enriched, and up-to-date provision of information on the food services available to older adults. In this regard, and as part of designing the marketplace's knowledge base, we present a first draft of the *Food Provider Ontology*, which guides the description of food providers and their offered food delivery services according to a number of characteristics capturing various aspects of diet and nutrition as well as other food-related aspects, to allow accommodating older adults' nutritional requirements, context and wishes.

2.2.1. Food Provider Ontology Design

The Food Provider Ontology captures knowledge about the domain of food providers and their available food delivery services. This ontology is depicted in Figure 2, illustrating the concepts in the domain and the relationships that hold between those concepts.

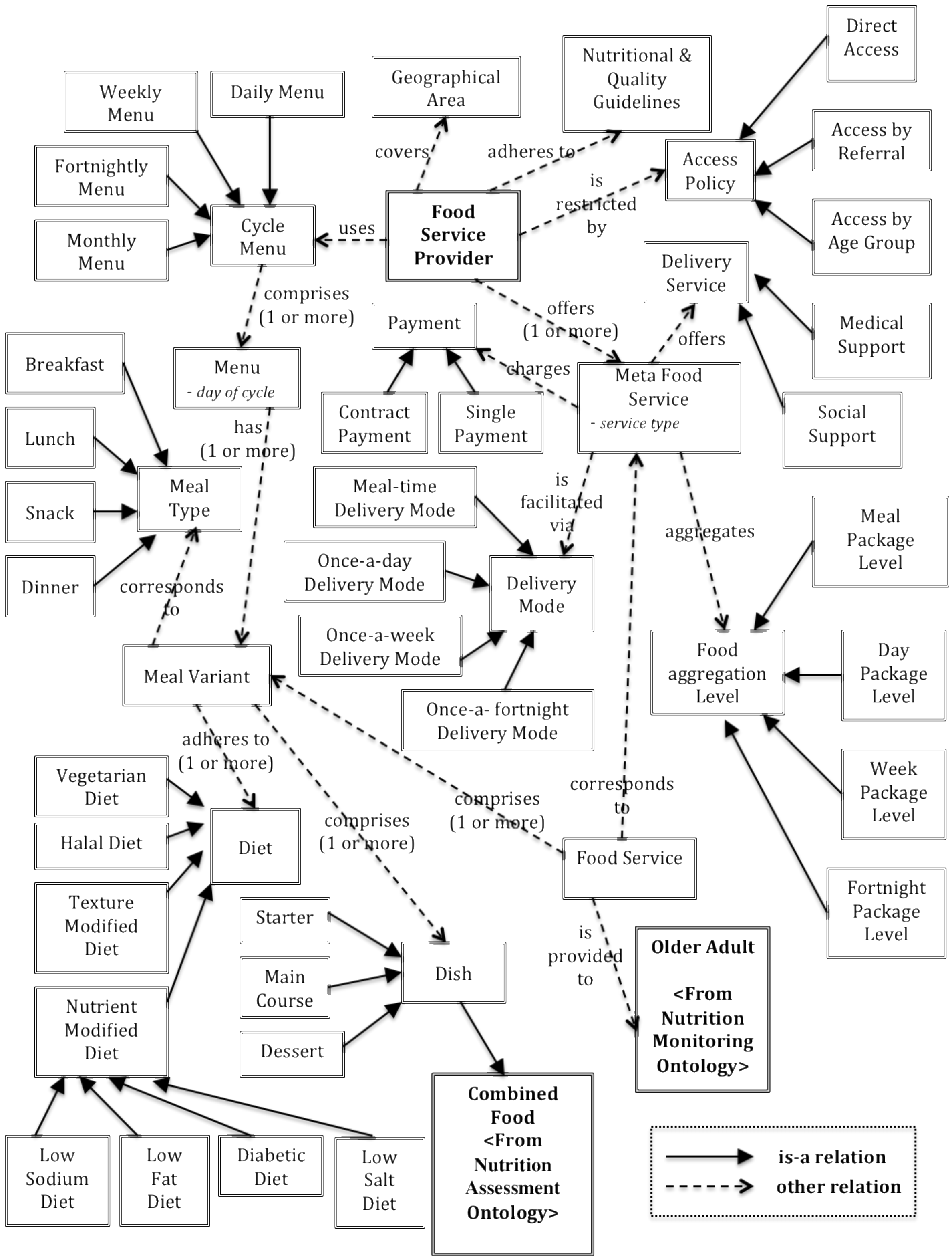

Figure 2. Food Provider Ontology

Food Service Provider is an entity providing access to nutritious food for older adults, either directly or via a third-party contractor. Such a provider could be a

Commercial Supplier, a *Community Agency*, a *Charity Organization*, etc. For example, a recent survey reveals a considerable number of food delivery services being offered for older adults across Scotland by local authorities [2].

Providers may operate over a limited *Geographical Area*. Moreover, access to their services may be available to everyone (*Direct Access*), be restricted to a particular age group (*Access by Age Group*), or requires a formal assessment or professional referral (*Access by Referral*).

Preferably, to ensure food safety and quality, providers should be able to indicate the *Nutritional and Quality Guidelines* that they adhere to. Examples of such guidelines include [2]: the National Association for Care Catering Standards, Caroline Walker Trust guidelines, Health and Nutrition Act 2007, etc.

A provider is likely to adopt a *Cycle Menu*[1], which is a set of *Menu*s (each for a particular day of the cycle) designed to cover a fixed period of time and then repeated. Depending on the length of the cycle chosen by the provider, such *Cycle Menu* could be a *Daily Menu* (repeated every day), a *Weekly Menu* (repeated every week), a *Fortnightly Menu* (repeated every two weeks), a *Monthly Menu* (repeated every month), etc. Normally, a longer menu cycle would be preferred by older adults to obtain a greater food variety. Each *Menu* comprising a *Menu Cycle* may in turn be composed of a number of sub-menus, corresponding to different *Meal Type*s, including a *Breakfast* meal, a *Lunch* meal, a *Snack* meal, and a *Dinner* meal.

To ensure variation and suitability to different dietary requirements, a number of *Meal Variant*s may be made available by the provider for each meal type in the menu, covering various types of *Diet* with specific constraints, including a *Vegetarian Diet*, a *Halal Diet*, a *Nutrient Modified Diet* (e.g. a *Low Sodium Diet*, a *Low Fat Diet*, a *Diabetic Diet*, a *Low Salt Diet*, etc.), a *Texture Modified Diet*, and other options.

Typically, each *Meal Variant* would consist of a number of *Dish*es (e.g. a *Starter*, a *Main Course*, and a *Dessert*). The characteristics of each dish, including the nutritional

[1] http://www.nfsmi.org/documentlibraryfiles/PDF/20071128104309.pdf

aspects and various other food-related features (such as ingredients, cuisine, flavor, etc.), are captured via utilizing the *Nutrition Assessment Ontology* of the Nutrition Care Process (see Deliverable D3.1), thus enabling diet-aware and preference-based selection of meals for older adults.

Given the cycle menu, the provider could offer a range of service delivery schemes, denoted in the ontology as *Meta Food Service,* to older adults. Each scheme is characterized by a number of features, including *Food Aggregation Level*, *Food Service Type*, *Payment Option*, *Delivery Mode*, and additional *Delivery Service*. Possible food aggregation levels include *Meal Package Level*, *Day Package Level*, *Week Package Level*, and *Fortnight Package Level*. Possible service types include *Hot Food Service* and *Frozen Food Service*. Possible payment options include *Single Payment Option* and *Contract Payment Option*. Possible delivery modes include *Meal-time Delivery mode*, *Once-a-day Delivery Mode*, *Once-a-week Delivery Mode*, and *Once-a-fortnight Delivery Mode*. Finally possible additional services on delivery could include *Medical Support* and *Social Support*.

Note that not every combination of the above feature options would form a valid service delivery scheme. For example, delivering a week food package in one go (i.e. once-a-week delivery mode) is better achieved through the frozen food service type, while meal-time and once-a-day delivery modes for such a package level would facilitate the hot food type.

Each *Food Service* is instantiated according to an available scheme, and concerns the delivery of a specific food package, comprising particular meal variants, to an older adult.

2.2.2. Ontology Implementation

For the purpose of implementing the Food Provider Ontology, we utilize the Web Ontology Language (OWL), a standard ontology language from the World Wide Web Consortium (W3C)[2]. We build such an OWL ontology using the Protégé tool[3], storing the resulting ontology file in ‘‘foodprovider.owl’’.

[2] http://www.w3.org/TR/owl-ref/
[3] http://protege.stanford.edu/

OWL ontology consists of Individuals, Classes, and Properties (Object Properties and Data Properties). Individuals are the objects of the domain. In our implementation, we maintain such objects through a relational database, which is mapped to the ontology via defining an appropriate mapping configuration, file ‘‘marketplacemapping.ttl’’ (see Section 3.2.2 of Deliverable D3.1 for more details on the mapping between a database and ontology). The Classes and the Properties of the Food Provider Ontology are detailed next.

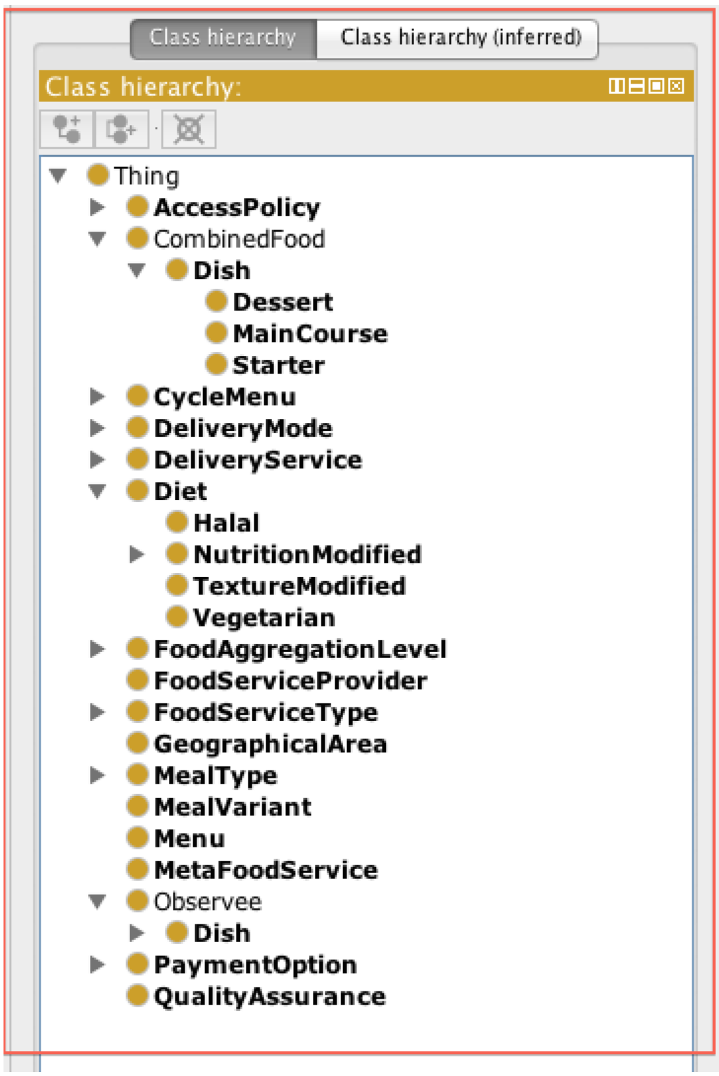

Figure 3. Class Hierarchy Tree for Food Provider Ontology

Ontology classes are the main building blocks of OWL ontology. They describe the concepts of the domain (each class characterizes a set of individuals in the domain). Classes are organized within class hierarchies, with a sub-class denoting a more specific concept than its super-class. Figure 3 shows the class hierarchy tree for the Food Provider Ontology corresponding to the concepts discussed in Section 2.2.1 (note that class *Thing* in this tree represents the set of all individuals).

Object Properties define the various types of relationships that exist between individuals in the domain.

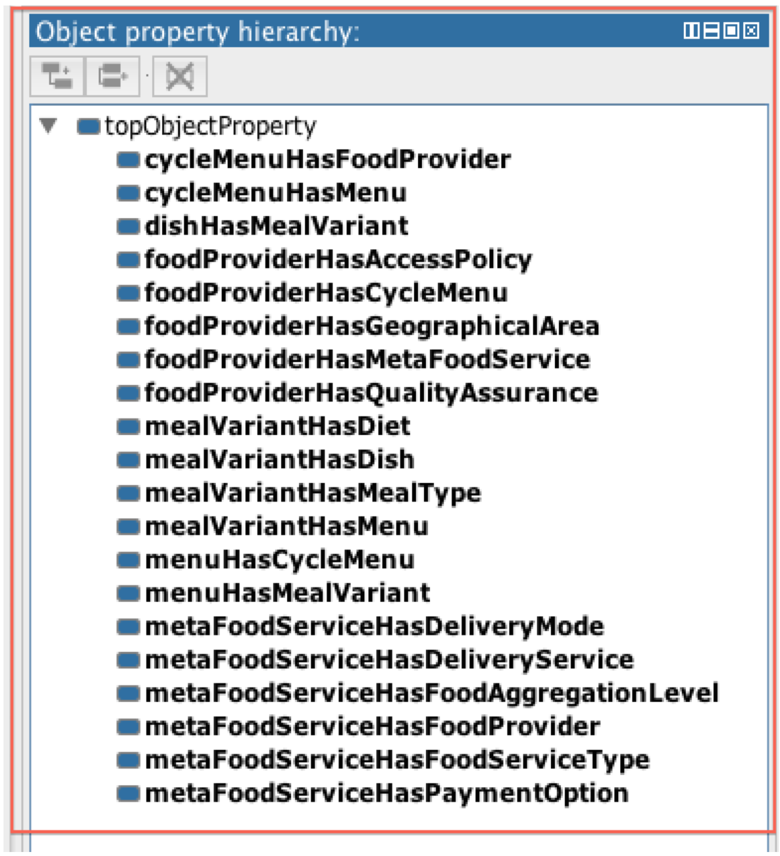

Figure 4. Object Properties for Food Provider Ontology

Each Object Property is associated with a domain and range, and links an individual from the domain defined class or concept to an individual from the range defined class or concept. Moreover, each Object Property may also have a corresponding inverse property. Figure 4 shows the Object Properties of the Food Provider Ontology, with their details being provided in Table 1.

Table 1. Object Property Details of Food Provider Ontology

Object Property	Domain	Range	Inverse Property
foodProviderHasAccessPolicy	FoodServiceProvider	AccessPolicy	
foodProviderHasCycleMenu	FoodServiceProvider	CycleMenu	cycleMenuHasFoodProvider
foodProviderHasGeographicalArea	FoodServiceProvider	GeographicalArea	
foodProviderHasMetaFoodService	FoodServiceProvider	MetaFoodService	metaFoodServiceHasFoodProvider
foodProviderHasQualityAssurance	FoodServiceProvider	QualityAssurance	
cycleMenuHasFoodProvider	CycleMenu	FoodServiceProvider	foodProviderHasCycleMenu
cycleMenuHasMenu	CycleMenu	Menu	menuHasCycleMenu
menuHasCycleMenu	Menu	CycleMenu	cycleMenuHasMenu
menuHasMealVariant	Menu	MealVariant	mealVariantHasMenu
mealVariantHasMenu	MealVariant	Menu	menuHasMealVariant
mealVariantHasMealType	MealVariant	MealType	
mealVariantHasDiet	MealVariant	Diet	
mealVariantHasDish	MealVariant	Dish	dishHasMealVariant
dishHasMealVariant	Dish	MealVariant	mealVariantHasDish
metaFoodServiceHasFoodProvider	MetaFoodService	FoodServiceProvider	foodProviderHasMetaFoodService
metaFoodServiceHasDeliveryMode	MetaFoodService	DeliveryMode	
metaFoodServiceHasDeliveryService	MetaFoodService	DeliveryService	
metaFoodServiceHasFoodAggregationLevel	MetaFoodService	FoodAggregationLevel	
metaFoodServiceHasServiceType	MetaFoodService	FoodServiceType	
metaFoodServiceHasPaymentOption	MetaFoodService	PaymentOption	

Data Properties define the various features and characteristics of individuals and classes. In particular, a Data Property links an individual from the *domain* Class to data values. Figure 5 shows the Data Properties of the Food Provider Ontology, with their details being provided in Table 2.

Table 2. Data Property Details of Food Provider Ontology

Data Property	**Domain**	**Range**
foodProviderHasId	FoodServiceProvider	long
foodProviderHasName	FoodServiceProvider	string
accessPolicyHasId	AccessPolicy	long
geographicalAreaHasId	GeographicalArea	long
geographicalAreaHasRegion	GeographicalArea	string
geographicalAreaHasCity	GeographicalArea	string
geographicalAreaHasCountry	GeographicalArea	string
qualityAssuranceHasId	QualityAssurance	long
qualityAssuranceHasDescription	QualityAssurance	string
cycleMenuHasId	CycleMenu	long
cycleMenuHasCycleLength	CycleMenu	long
cycleMenuHasStartDate	CycleMenu	dateTime
cycleMenuHasValidityDate	CycleMenu	dateTime
menuHasId	Menu	long
menuHasDayOfCycle	Menu	long
menuHasDayOfWeek	Menu	string
mealVariantHasId	MealVariant	long
mealVariantHasSpecificDiet	MealVariant	string
mealTypeHasId	MealType	long
dietHasId	Diet	long
dietHasName	Diet	string
dishHasId	Dish	long
dishHasName	Dish	string
foodAggregationLevelHasId	FoodAggregationLevel	long
foodServiceTypeHasId	FoodServiceType	long
paymentOptionHasId	PaymentOption	long
deliveryModeHasId	DeliveryMode	long
deliveryServiceHasId	DeliveryService	long
metaFoodServiceHasId	MetaFoodService	long

Figure 5. Data Properties for Food Provider Ontology

2.3. Food Service Reliability Management

In what follows, we first analyze a number of relevant reliability dimensions for food delivery services, and then introduce the first draft of the *Reliability Monitoring Ontology*, which will guide the monitoring and collection of past performance data on food services, and assessing these services with respect to the reliability aspects proposed.

2.3.1. Reliability Dimensions of a Food Service

Multiple perspectives could be considered for assessing the reliability of food delivery services, including computational reliability, product reliability, and fault tolerance. We elaborate on these aspects in more detail next.

Computational reliability is the reliability of the software component encapsulating the service (i.e. the software component through which the service is made available online), and corresponds to the probability that this component will respond to the request successfully. Computational reliability could be either: (i) determined by the service provider; or (ii) estimated from historical information, e.g. as the ratio of successful transactions (order placements) over total number of transactions.

Product reliability refers to the various features of the food product that the service delivers, including food quality aspects (such as taste, smell, etc.), nutrition aspects, and delivery aspects (such as delivery time, delivery staff, etc.). These features may be initially specified by the service provider, with more accurate (correct) estimates being *learned* over time from collected actual performance information (see Table 3 for some examples).

Besides previous experience data, other factors could be taken into consideration when learning a product's features (or assessing the trustworthiness of its advertised properties), especially when the previous experience with the product is not sufficient:

- Experience with other products from the same provider. For example, poor packaging or delays in delivery could be anticipated when ordering a meal from a restaurant if frequently experienced before with other meals from the same restaurant.
- Semantic relations between products, such as:
    - similarity relation (same food categories or meals with similar ingredients are likely to share similar nutrition features)
    - aggregation relation (the nutrition features of a compound product could be derived from the available nutrition information of its comprising components)

These relations can be particularly useful to utilize in the absence of nutrition data from service providers.

Table 3. Collecting Actual Performance Data

Feature	Performance Collection
food quality aspects	• Older adult/informal carer feedback; • Chemical analysis of food; • Older adult health monitoring, e.g. reaction to food, etc.
nutrition aspects	• Chemical analysis of food; • Older adult health monitoring, e.g. blood tests, etc.
delivery aspects	• Monitoring agent, which measures the time between order placement and delivery to the required destination, e.g. utilising door sensors, etc. • Older adult/informal carer feedback.

Fault Tolerance incorporates *Service Replaceability* and *Service Compensability*.

Service Replaceability is the availability of potential backups in the case of service failure (e.g. the product could not be delivered; the product does not meet its expectations, etc.). A service is considered a backup for another service if it provides similar features and capabilities, e.g. similar food category, similar ingredients, similar nutrition properties, similar price, etc. Reasoning about service replaceability when planning the composition of food services (i.e. during daily/weekly/monthly dietary menu planning) allows producing more reliable compositions, avoiding unrecoverable situations in the case of component service failures.

Service Compensability is the ability to undo the service (i.e. remove the service's effects) once invoked or executed. For food services, the *undo* action could include refunding the cost (either partially or fully), returning the product to the provider, etc. Compensable services are more reliable than non-compensable ones due to several reasons, including: avoiding inconsistent states in the case of service failure; and facilitating the rollback of already executed compositions (i.e. already ordered menu plans) to earlier execution points, from which satisfactory (or better) replacements can be found.

2.3.2. Reliability Monitoring Ontology

Based on above, we present the first draft of the Reliability Monitoring Ontology, which will guide the monitoring and collection of past performance data on food services with respect to the reliability dimensions proposed. The Reliability Monitoring Ontology is depicted in Figure 6, with the concepts involved being outlined below.

A *Reliability Dimension is* a feature of interest to be assessed for a food service (which may or may not be initially specified by the provider). It captures the aspects discussed in Section 2.3.1. The *Food Quality Aspect* includes *Portion Size*, *Presentation*, *Smell*, *Taste*, *Texture*, and *Value for Money*. The *Delivery Aspect* covers *Delivery Package* (with respect to *Expectation Meeting*, *Hygiene*, *Labeling*, and *Temperature*), *Delivery Staff* (with respect to offered *Professional Help* and *Social Interaction*), and *Delivery Time*.

Two *Types* are possible for such features: *Objective* (e.g. the *Delivery Time* feature), and *Subjective* (e.g. the *Taste* feature, which could be rated differently by users depending on their background, preferences, etc.).

The *Domain* (i.e. the set of possible values) of each of such dimensions can be either a *Numerical Domain* or a *Categorical Domain*. The former is further categorised into a *Discrete Domain* and a *Continuous Domain*, while the latter is categorised into a *Nominal Categorical Domain* and an *Ordinal Categorical Domain*, consisting of a number of corresponding *Domain Items*. A *Domain* might be associated with a *Direction*, specifying the ordering of values in this domain (applicable to numerical and ordinal categorical value domains). Such direction could be either *Increasing* (i.e. the higher the value the better) or *Decreasing* (i.e. the lower the value the better).

An *Observation* evaluates a particular *Reliability Dimension* for a particular *Observe (e.g. a Food Service, a Dish)*, and reports the value encountered for this feature (from the corresponding *Domain)* at a particular time. Such an observation is collected from a particular *Observer*, which could be either a *Human User* (i.e. the older adult or their carer) or a *Sensor*. The collected observations over time on food services can be utilized for the purpose of providing more accurate value estimates of service features, and enabling more informative and reliable service selection and recommendation for older adults.

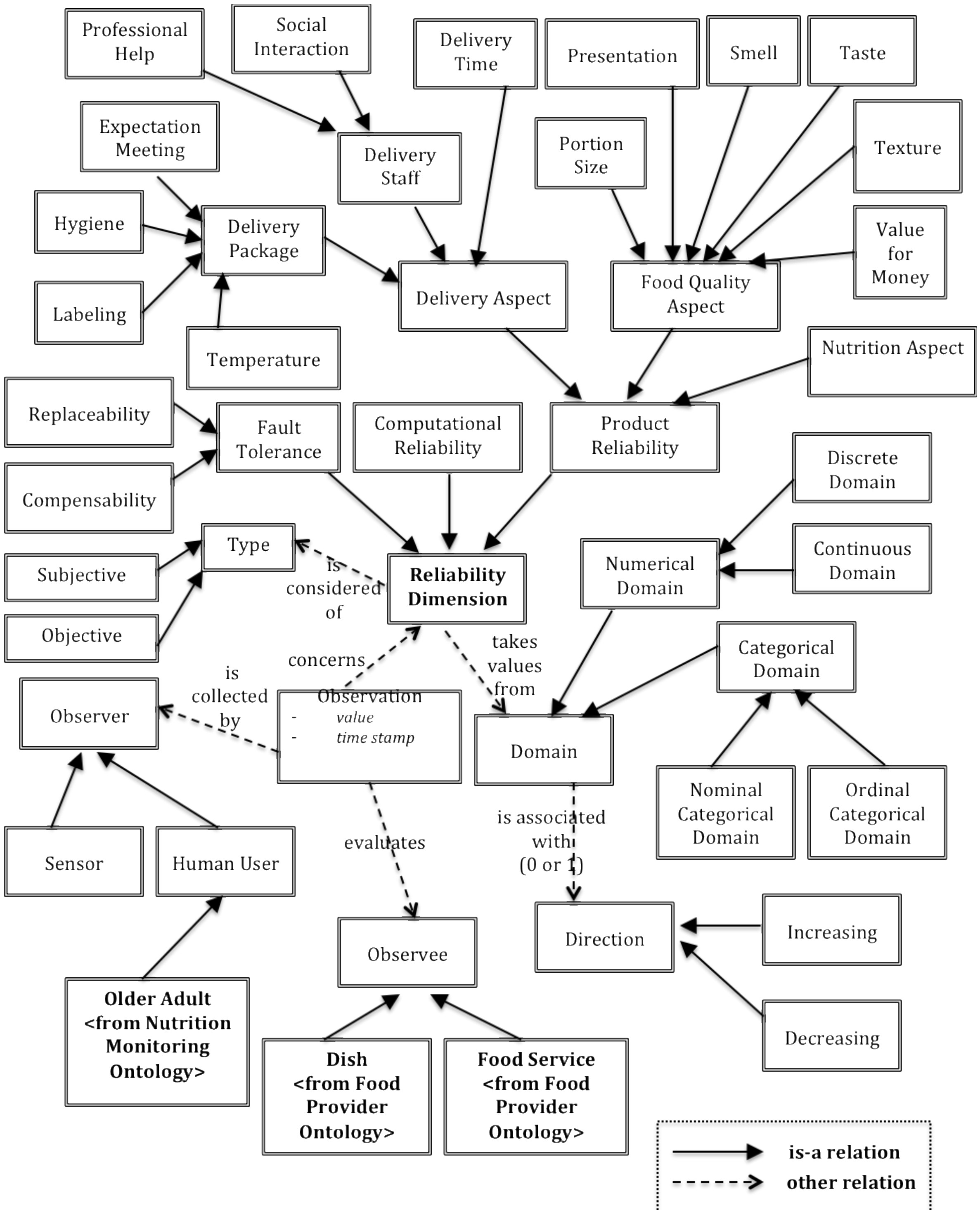

Figure 6. Reliability Monitoring Ontology

2.3.3. Ontology Implementation

Similarly to the Food Provider Ontology, we implement the Reliability Monitoring Ontology using OWL and Protégé tool, storing the resulting ontology file in ‘‘ReliabilityMonitoring.owl’’.

The class hierarchy tree for the Reliability Monitoring Ontology is depicted in Figure 7; the Object Properties and their details are provided in Figure 8 and Table 4, respectively; and finally the Data Properties and their details are showed in Figure 9 and Table 5.

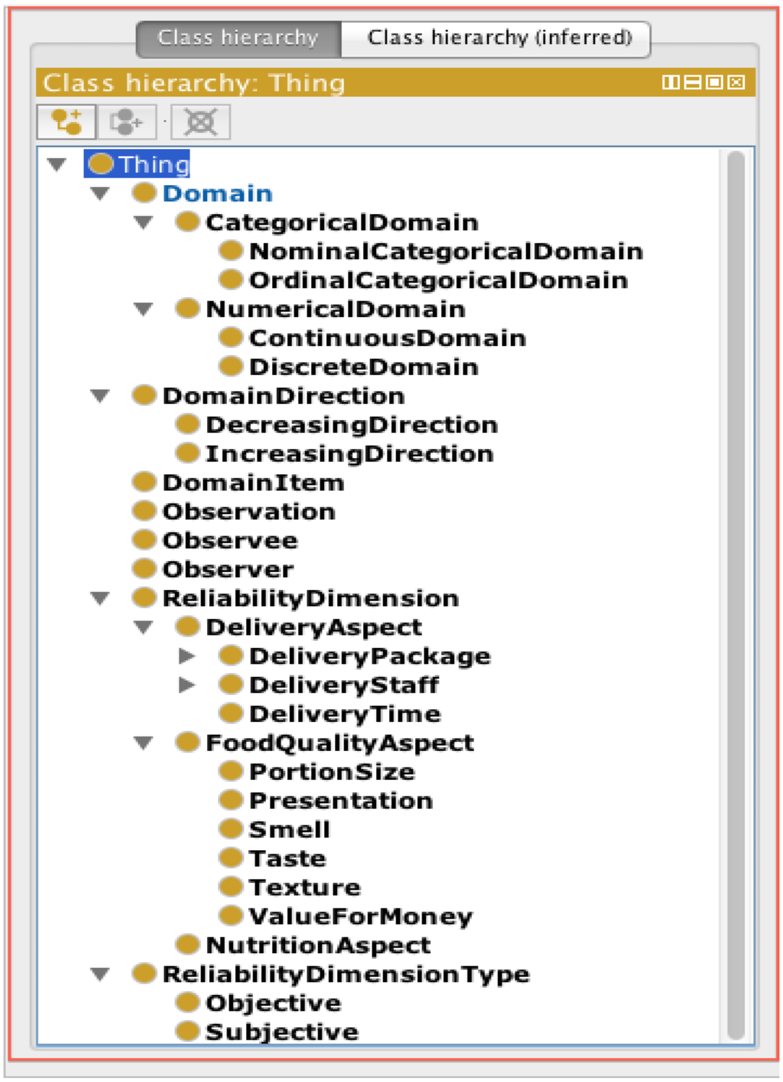

Figure 7. Class Hierarchy Tree for Reliability Monitoring Ontology

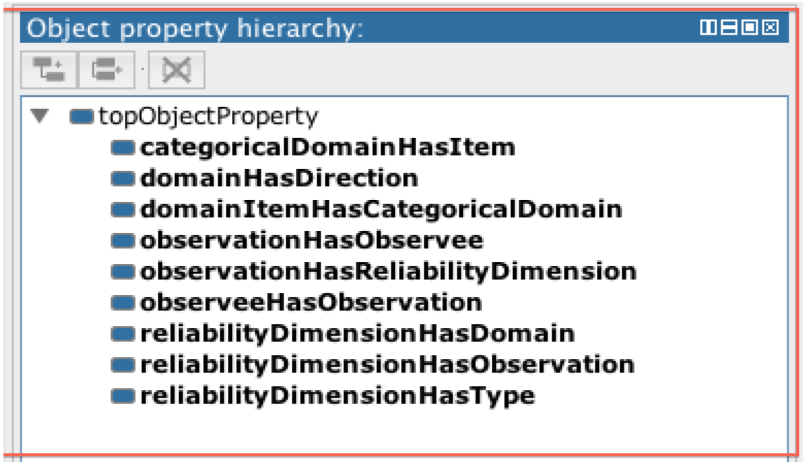

Figure 8. Object Properties for Reliability Monitoring Ontology

Table 4. Object Property Details of Reliability Monitoring Ontology

Object Property	Domain	Range	Inverse Property
reliabilityDimensionHasType	ReliabilityDimension	ReliabilityDimensionType	
reliabilityDimensionHasDomain	ReliabilityDimension	Domain	
reliabilityDimensionHasObservation	ReliabilityDimension	Observation	observationHasReliabilityDimension
domainHasDirection	Domain	DomainDirection	
categoricalDomainHasItem	CategoricalDomain	DomainItem	domainItemHasCategoricalDomain
domainItemHasCategoricalDomain	DomainItem	CategoricalDomain	categoricalDomainHasItem
observationHasObservee	Observation	Observee	observeeHasObservation
observationHasReliabilityDimension	Observation	ReliabilityDimension	reliabilityDimensionHasObservation
observeeHasObservation	Observee	Observation	observationHasObservee

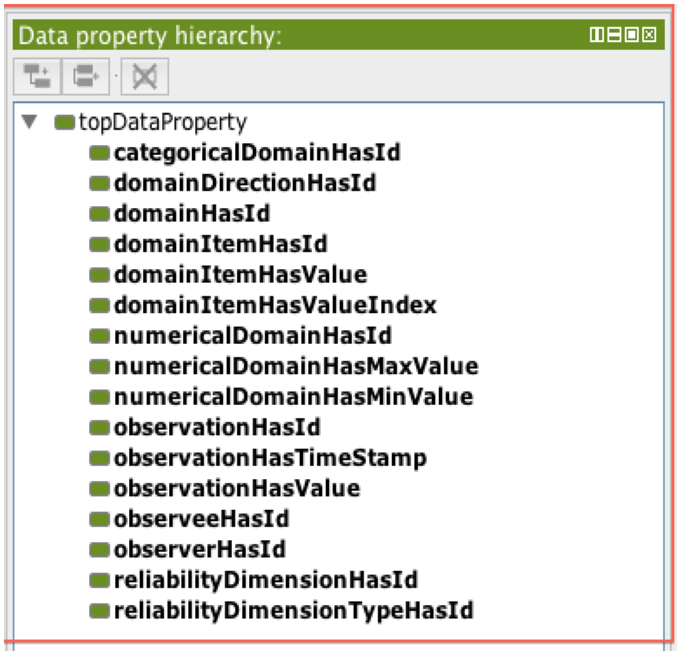

Figure 9. Data Properties for Reliability Monitoring Ontology

Table 5. Data Property Details of Reliability Monitoring Ontology

Data Property	Domain	Range
reliabilityDimensionHasId	ReliabilityDimension	long
reliabilityDimensionTypeHasId	ReliabilityDimensionType	long
domainHasId	Domain	long
domainItemHasId	DomainItem	long
domainItemHasValue	DomainItem	string
domainItemHasValueIndex	DomainItem	long
domainDirectionHasId	DomainDirection	long
categoricalDomainHasId	CategoricalDomain	long
numericalDomainHasId	NumericalDomain	long
numericalDomainHasMinValue	NumericalDomain	double
numericalDomainHasMaxValue	NumericalDomain	double
observationHasId	Observation	long
observationHasTimeStamp	Observation	dateTime
observationHasValue	Observation	string
observeeHasId	Observee	long
observerHasId	Observer	long

2.4. Marketplace Abstract Knowledge Model

In this section, we present a formal representation of the marketplace outlined in the previous sections. This formal model captures the knowledge needed for facilitating optimal selection and composition of food services for older adults (e.g. by the diet-aware food service composition engine), which satisfy their complex dietary recommendations and food preferences. Specifically, knowledge of food services maintained by the marketplace comprises four components, abstract service hierarchy, quality meta-model, service model, and reliability model, which are detailed below (note that this model is inspired by the multi-granularity service model proposed by Barakat et al. [3]).

2.4.1. Abstract Service Hierarchy

A food service, registered by a provider in the marketplace, encapsulates the functionality of delivering a food product to an older adult. Among other features, food services may vary in granularity, i.e. the food product delivered by the service could be a particular dish, a 3-course meal, or even an entire daily or weekly menu. Knowledge of such multiple granularity levels enables the discovery and utilization of the corresponding available food services, and can be represented as a hierarchy of abstract food services, where each abstract service can be further decomposed into finer-grained services. The leaf nodes correspond to atomic abstract services (e.g. a single dish), while the root represents the highest level of abstraction (e.g. a week or a fortnight menu package).

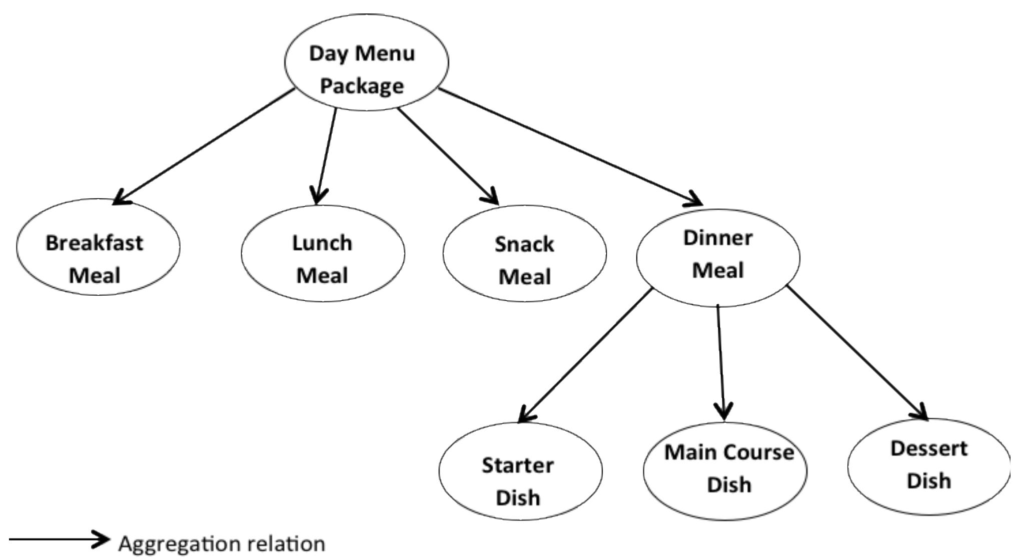

Figure 10. Abstract Service Hierarchy

For example, part of this hierarchy is depicted in Figure 10. As can be seen, a day menu package can be divided into a number of meal packages, with each further consisting of a number of dishes.

Formally, the abstract service hierarchy can be defined as a tuple, $(AFS, Aggr)$, where AFS is a finite set of abstract food services; and $Aggr$ is an aggregation relation between abstract services, hierarchically decomposing coarse-grained abstract services into finer-grained ones, with $(as_i, as_j) \in Aggr$ indicating that as_i is the aggregating (parent) abstract service while as_j is the aggregated (child) one. For example, the hierarchy of Figure 10 can be defined as follows:

$$AFS = \left\{ \begin{matrix} \textit{day menu package, breakfast meal, lunch meal, snack meal, dinner meal,} \\ \textit{starter dish, main course dish, dessert dish} \end{matrix} \right\}$$

$$Aggr = \left\{ \begin{matrix} \textit{(day menu package, breakfast meal),} \\ \textit{(day menu package, lunch meal),} \\ \textit{(day menu package, snack meal),} \\ \textit{(day menu package, dinner meal),} \\ \textit{(dinner meal, starter dish),} \\ \textit{(dinner meal, main course dish),} \\ \textit{(dinner meal, dessert dish)} \end{matrix} \right\}$$

2.4.2. Quality Meta-Model

To allow accommodating older adults' nutritional requirements, context and wishes, available food services need to be described according to number of quality characteristics, capturing various aspects of diet and nutrition, as well as other food-related aspects including food quality, type of service, delivery, cost, geographical coverage, etc. Formally, knowledge of such quality attributes comprises a tuple, $(ANAME, adom)$, where:

- $ANAME$ is the set of quality attributes describing food services (corresponding to concepts from the Food Provider and Reliability Monitoring Ontologies):

$$ANAME = \left\{ \begin{matrix} \textit{salt value, energy value, … , flavour, smell, … ,} \\ \textit{diet type, ingredients, delivery time, cost, …} \end{matrix} \right\}$$

- $adom: ANAME \rightarrow 2^{AVAL}$ is a domain function, mapping each quality attribute to its corresponding domain (possible values of this attribute), where $AVAL$ is the union of the domains of all quality attributes. For example:

  $adom(salt\ value) = \mathbb{R}^+$

  $adom(flavour) = \{sweet, sour, spicy, \dots\}$

  $adom(diet\ type) = \{general, diabetic, low\ sodium, \dots\}$

  $adom(ingredients) = 2^{FOODITEM}$, where $FOODITEM$ is the set of Food concepts defined in the Food Ontology.

2.4.3. Service Model

The space of all food services (available through the marketplace), along with their descriptive metadata, could be modeled as a tuple $(FS, fslevel, fsdesc)$ where:

- FS represents the set of all food services available for older adults, which are registered in the marketplace by various providers. As stated earlier, by food service, we refer to any food package made available for delivery. Note that, depending on the provider policy, the components of a composite food package may also be regarded as individual food services.
- $fslevel: FS \rightarrow AFS$ is the granularity level function, which maps each actual food service to its corresponding abstraction level (i.e. abstract food service).
- $fsdesc: FS \times ANAME \rightarrow AVAL$ is the food service description function, which assigns to each food service its values for the various quality characteristics specified in the Quality Meta-Model, such that $\forall s \in FS$, $\forall a \in ANAME$, $fsdesc(s, a) \in adom(a)$. These values are either made available by the provider, or estimated automatically since some quality values of a composite food package can be calculated through an appropriate aggregation of the respective quality values of the comprising components. Such aggregation, for instance, could be the summation function (e.g. for the salt value attribute) or the union function (e.g. for the ingredients attribute).

2.4.4. Reliability Model

Based on the performance history of food services, maintained within the service reliability manager, the reliability score of a food service with respect to a quality attribute (reliability dimension) can be estimated by applying some aggregation (e.g. a time-weighted average) on the available past ratings. Formally, the reliability scores corresponding to food services can be defined as a function:

$$relscore: FS \times ANAME \cup \{all\} \rightarrow RSDOM$$

where $relscore(s, a)$ is the reliability score of food service s with respect to quality attribute a, while $relscore(s, all)$ is the overall reliability score of food service s, and $RSDOM$ is the domain of reliability scores.

3. Hybrid bio-inspired techniques for food services dynamic discovery and composition

The selection of a food services combination which satisfies complex criteria, such as the older adult recommended diet or preferences for a certain type of food, is not a trivial task and cannot be addressed using conventional techniques. There are large numbers of food delivery services available on the market offering various types of food, which makes the finding of the optimal combination an NP-hard problem that cannot be solved in reasonable time using conventional techniques. The DIET4Elders solution is to model the problem as a combinatorial optimization problem and solve it using hybrid bio-inspired techniques that should combine the strength elements of different bio-inspired meta-heuristics. The advantage of such hybrid techniques is that they allow identification of the optimal or a near-optimal solution without processing the entire search space as opposed to a classical exhaustive search strategy.

3.1. Technical approach overview

In order to identify the optimal combination of food delivery services using bio-inspired meta-heuristics, the following steps need to be performed (see Figure 11): (1) the optimization problem needs to be formally represented and the search space defined, (2) the concepts of the optimization problem must be mapped to the meta-heuristic concepts and appropriate fitness functions need to be defined, and (3) the meta-heuristic's algorithm needs to be adapted/enhanced according to the optimization problem being solved.

The *first step* is tackled by defining the search space as the set of food services available in the cloud that are semantically annotated with information regarding various aspects of nutrition (such as organic composition, energetic values, calories etc.) and food quality. As stated in Section 2, the services will be semantically annotated with such information during the food service registration and monitoring within the marketplace, using concepts defined in the Food Provider Ontology and Reliability Monitoring Ontology. The food service search space is formally defined in Section 2.4.3. The nutrition and diet related knowledge, which will influence the selection and combination processes, will be provided by means of the Nutrition Care Process Ontology, while the users' preferences and needs will be captured by means of user profiles and used to refine the food selection process.

The *second step* aims to map the concepts of the optimization problem to the concepts of the chosen meta-heuristic and to provide an appropriate formal representation of these concepts in a way to enable a low processing overhead. On the search space, we will investigate and define a set of fitness functions (e.g. a fitness function evaluating the number of calories which needs to be minimized based on the food descriptions provided by the food delivery services) that evaluate the quality of a food service combination according to the criteria (e.g. number of calories) established in the older adult recommended diet.

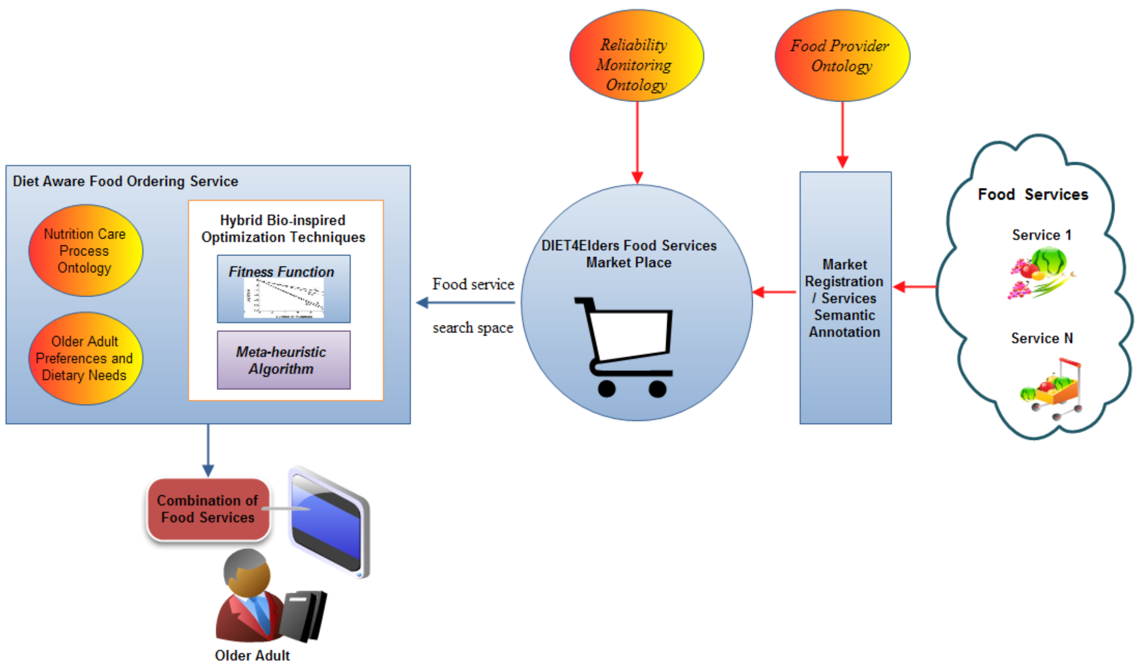

Figure 11. Food Services Dynamic Discovery and Composition

The *third step* involving the adaptation/enhancement of the meta-heuristic's algorithm implies the algorithm specialization according to the problem being solved and additionally including some supplementary processing steps to provide a balance between exploration and exploitation. For example, if the Ant Colony Optimization meta-heuristic is considered, in our problem of selecting the optimal combination of food delivery services we will have a number of intelligent agents (i.e. implementing the ants behavior) that cooperate with each other by indirectly exchanging information (i.e. like the ant's pheromone in nature) to identify the optimal or a near-optimal food delivery service combination (i.e. the ant's food source in nature) encoded in the search space (i.e. the environment in which real ants live).

In the next subsection we will detail four hybrid bio-inspired techniques that we have developed to cope with the specificity of diet aware food services selection and composition.

3.2. Hybrid Invasive Weed Heuristic

The invasive weed technique is based on the phenomenon of colonization of invasive weeds in nature [4]. In nature, weeds profit of the unused resources from the soil and grow until they become mature weeds that produce seeds. The number of seeds of each weed depends on the quantity of resources consumed and on the ability to adapt to the environment (i.e. the fitness of the weed in the colony). The seeds produced are scattered over the ground and grow until they become mature weeds. The process is repeated until the unused resources from the soil are totally depleted.

Starting from the above presented behavior we have defined a hybrid invasive weed technique for generating optimized food combinations starting from a given user profile, a diet recommendation, and a set of food services available in the marketplace.

We have started by mapping the concepts of the Invasive Weed Optimization meta-heuristic onto the concepts of generating optimal combination of food services that meet various constraints, as follows:

(i) a weed (i.e. a solution) is represented as a set of food offers from different food service providers representing the food items for breakfast, snacks, lunch, or dinner. It is described using the concept of Food describing in in Nutrition Care Process Ontology which has different granularities Basic Food, Combined Food, Dish, etc.

(ii) the seed of a weed represents a new set of food offers obtained by performing a random mutation of the old set of food offers in a number of mutation points. A mutation is generated by replacing a food offer in a solution with another one available in the marketplace according to the

user's wishes and needs, the recommended diet constraints and nutrition related knowledge;

(iii) the colony represents the whole population of weeds (i.e. sets of food offers);

(iv) the fitness value (calculated based on a fitness function) represents how good a solution is, regarding the given constraints.

In our approach, a weed is formally represented as:

$$sol = \{ FI_b, FI_{s1}, FI_l, FI_{s2}, FI_d \} \quad (1)$$

where FI_b represents the set of food items that can be eaten at breakfast, FI_{s1} and FI_{s2} are sets of food items that can be eaten at snacks, FI_l at lunch, and FI_d at dinner, during one day.

The set FI_b of food items that can be eaten at breakfast is defined as:

$$FI_b = \{FoodItem_{b1}, FoodItem_{b2}, \dots, FoodItem_{bn}\} \quad (2)$$

In relation (2), $FoodItem_{b2}$ is a food item that can be eaten at breakfast during one day and *n* is the number of food items that can be consumed at breakfast during one day. The other components of a solution corresponding to the snacks, lunch and dinner are defined similarly to FI_b.

Having the formalism defined above, the next step is to define a fitness function based on which the quality of a solution (i.e. a combination of food items) can be assessed. We have defined a fitness function which takes into consideration the following aspects as defined by the DIET4Elders food services reliability ontology: nutrition aspect, food quality aspect and delivery aspect.

The fitness function used to evaluate the quality of a solution (i.e. weed) is defined as:

$$Fitness\ (sol) = \alpha_1 * F_{nutrition}(sol) + \alpha_2 * F_{quality}(sol) + \alpha_3 * F_{delivery}(sol) \quad (3)$$

where: $\alpha_i \epsilon [0,1]$ are the weighting coefficients and $\sum_{i=1}^{3} \alpha_i = 1$;

The **nutrition related aspect** of the fitness function is captured using relation (4) and evaluates the nutritional features (i.e. calories, proteins, lipids, carbohydrates, etc.) of a solution (i.e. a daily menu).

$$F_{nutrition}(sol) = \sum_{i=1}^{n} \omega_i * f_i(sol) \tag{4}$$

In relation (4), $\omega_i \in [0,1]$, $\sum_{i=1}^{n} \omega_i = 1$, is the weight reflecting the importance of each nutritional feature as defined by a nutritionist while the f_i function evaluates a nutritional feature for the entire day menu solution (i.e. over all the food items comprising the solution):

$$f_i(sol) = \frac{optVal_i - \sum_{FoodItem_j \in sol} NF_i(FoodItem_j)}{optVal_i} \tag{5}$$

where: (i) $NF_i(FoodItem_j)$ is the value of a particular nutritional feature (i.e. calories, proteins, lipids, carbohydrates, etc.) for a food item contained in a daily menu solution; (ii) $optVal_i$ is the optimal value for a particular nutritional feature for a day; this value is provided by nutritiotions or it can refer to the Recommended Daily Intake values.

The value of a particular nutritional feature for a specific food item is either made available by the food provider, or is calculated from the ingredients of the food item taken from the food sub-ontology as defined in the DIET4Elders Nutrition Care Process Ontology.

Additionaly, we have defined a set of nutrition or dietary constraints ($C_j(y)$) which will be used for eliminating those solutions that are not feasible for an older adult specific condition (i.e. food with sugar for a diabetic). The following relation is used:

$$C_j(y) = \sum_{y_i \in sol} \sum_{k=1}^{n(y_i)} q_{ik} * p_{kj} \tag{6}$$

where: (*i*) $n(y_i)$ is the number of ingredients in food item y_i; (*ii*) q_{ik} is the quantity of ingredient k contained into a food item y_i, and (*iii*) p_{kj} is the percentage of nutrient j that is contained into ingredient k.

There may be other constraints in addition to the nutrition constraints such as some ingredient-related constraints due to older adult allergies or defined preferences which are also considered for reducing the search space before running the defined heuristic.

These constraints are satisfied if they take values in an interval of values $[min_j, max_j]$, where min_j is the minimum daily requirement for a particular nutrient j established by the nutritionist (e.g. sodium, iron, A, B1 or C vitamins), and max_j is the maximum daily requirement for nutrient j.

The **food quality aspect** of the fitness function is captured based on the user feedback on specific food items (for example by giving notes from 1 to five describing his/her experience of eating a specific food item ordered) offered by different food services available in the marketplace and using the reliability aware ontology provided concepts.

$$F_{quality}(sol) =$$

$$q_1 * \sum_{FoodItem_i \in sol} \frac{smell_{rate}}{MAX(smell_{rate})} + q_2 * \sum_{FoodItem_i \in sol} \frac{taste_{rate}}{MAX(taste_{rate})} + \ldots \quad (7)$$

Relation (7) evaluates the food quality related aspects for a solution as defined in the reliability monitoring ontology, aggregating these aspects over the food items comprising the solution ($q_1, q_{2\ldots}$ are weights showing the importance of a quality criteria).

The **delivery aspect** of a daily menu solution is referring to the delivery time. Relation (8) evaluates the delivery time associated to a solution and is computed as the maximum time among the delivery times of the solution components:

$$F_{delivery}(sol) = MAX_{FoodItem_i \in sol} DeliveryTime(FoodItem_i) \quad (8)$$

As we have already mentioned above the values of the main fitness function components describing the three main aspects of the solution as well as the values of the constraints are normalized.

3.2.1. Hybridization alternatives

This section presents the alternative hybridization methods that we have defined and used to improve the Invasive Weed Optimization meta-heuristics performance in terms of solution exploitation (i.e. efficient processing of promising solutions neighborhood) and exploration (i.e. efficient processing of new areas of the search space).

We have hybridization techniques based on Tabu Search and path relinking.

Tabu Search based hybridization

Our goal is to improve the search capabilities of the invasive weed meta-heuristic by means of long-term and short-term memory structures borrowed from Tabu search [5].

The **long-term memory** (see relation (9)) contains the history of food item replacements and the associated rewards and penalties and it is consulted each time a new solution is generated.

$$M_{long} = \{m_{ij} \mid m_{ij} = (FoodItem_i, FoodItem_j, reward)\} \quad (9)$$

In relation (9), $FoodItem_i$ is a food item that has been replaced in an intermediary solution by the food item $FoodItem_j$ (i.e. mutation process), while the $reward$ parameter is used to record rewards and penalties of the mutation process. The $reward$ value of a food item mutation is updated each time the specified replacement takes place in order to modify a solution. If the replacement improves the quality of a solution then the $reward$ value is increased (a reward is granted), otherwise the $reward$ value is decreased (a penalty is granted).

The **short-term memory** structure (see relation (10)) contains food item mutations and the number of future iterations in which those mutations cannot be used.

$$M_{short} = \{m_{ij} \mid m_{ij} = (FoodItem_i, FoodItem_j, noIterations)\} \quad (10)$$

In relation (10), the food item $FoodItem_i$ part of a daily menu solution has been replaced with another food item $FoodItem_j$ (during the mutation process) and $noIterations$ value defines number of next consecutive iterations in which the same mutation can't be made. The value of the $noIterations$ parameter is set dynamically according to the need for exploration versus exploitation. When rather exploration is needed then $noIterations$ is increased, otherwise $noIterations$ is decreased.

Path relinking based hybridization

Path relinking [6] is a search technique that exploits the search space by finding paths between two solutions (i.e. the initial solution and the guiding solution). The main idea of this technique is to obtain the guiding solution by applying a set of modification strategies on the initial solution. This way, each time a mutation is applied on the initial solution, the intermediate solution obtained will be more similar with the guiding solution and less similar with the initial one. Using this searching technique, new paths between the initial and the guiding solution are generated, and the solutions on these paths could be a source for generating new paths.

In our approach, the path relinking hybridization is used to generate new solutions by considering as initial solution the daily menu with the highest value of the fitness function from the ones generated in the current step and as guiding solution the daily menu with the highest value of the fitness function at the next step.

For generating new solutions (daily menus), we have used two path relinking based strategies: one which applies a crossover mutation operator to modify the initial solution towards the guiding solution, and another one which applies a Particle Swarm Optimization (PSO) [7] modification strategy.

In the **path relinking strategy based on crossover operator** the initial solution is updated until arriving at the current step optimal solution by iteratively performing a one point crossover (the crossover point being randomly chosen) between the two solutions (see Figure 12).

Initial solution: $sol_i = (foodItem_1, foodItem_2, foodItem_3, foodItem_4)$,

Guiding solution: $sol_g = (foodItem_1', foodItem_2', foodItem_3', foodItem_4')$

First Step

$sol_i' = (foodItem_1', foodItem_2, foodItem_3, foodItem_4)$,

$sol_g' = (foodItem_1, foodItem_2', foodItem_3', foodItem_4')$

Second Step

$sol_i'' = (foodItem_1', foodItem_2', foodItem_3, foodItem_4)$,

sol_g'' = *(foodItem$_1$, foodItem$_2$, foodItem$_3$', foodItem $_4$')*

Third Step

sol_i''' = *(foodItem$_1$', foodItem$_2$', foodItem$_3$', foodItem$_4$),*

sol_g''' = *(foodItem$_1$, foodItem$_2$, foodItem$_3$, foodItem$_4$')*

Figure 12. Example of using path relinking strategy based on crossover operator

In the **PSO-based path relinking strategy** the initial solution is updated by iteratively applying relation (11), until generating a new solution having a fitness value higher than the current globally optimal solution.

$$sol_{new} = (1-\beta) * sol_i + \beta * sol_g + Vector_r \qquad (11)$$

In relation (11), β represents a value in the interval (0, 1) used to compute the percentage of food items from the initial solution sol_i that will be substituted with new food items from the guiding solution sol_g, and $Vector_r$ is a randomly generated vector of 0's and 1's with a length equal to the length of the initial and guiding solution (1 indicates that the original food item from the initial solution is kept while 0 indicates that the original food item from the initial solution is replaced by the corresponding one from the guiding solution). Figure 13 presents a trace example of this approach.

Initial solution: sol_i = (*foodItem*$_1$, *foodItem*$_2$, *foodItem*$_3$, *foodItem*$_4$)

Guiding solution: sol_g = (*foodItem*$_1$', *foodItem*$_2$', *foodItem*$_3$', *foodItem*$_4$')

β = 0.7

First step

(1- **β**)* (*foodItem*$_1$, *foodItem*$_2$, *foodItem*$_3$, *foodItem*$_4$) +

β * (*foodItem*$_1$', *foodItem*$_2$', *foodItem*$_3$', *foodItem*$_4$') =

(*foodItem*$_1$, *foodItem*$_2$', *foodItem*$_3$', *foodItem*$_4$')

Second step

(*foodItem*$_1$, *foodItem*$_2$', *foodItem*$_3$', *foodItem*$_4$') + (1, 0, 0, 1) =

(*foodItem*$_1$, *, *, *foodItem*$_4$')

Figure 13. Example of using PSO-based path relinking strategy

3.2.2. Hybrid Invasive weed Optimization Algorithm

The Hybrid Invasive Weed Optimization algorithm takes as input the following parameters:

- *dietaryRec* – the older adult recommended diet,
- *foodOffers* **-** the set of food service offers available in the DIET4Elders marketplace,
- *personalProfile* **-** the personal profile of the older adult captured and stored by means of the DIET4Elders Monitoring Ontology,
- *initialPopSize* **-** the size of the initial population,
- *maxSeeds* **-** the maximum number of seeds,
- *minSeeds* - the minimum number of seeds,
- *noIt* **-** the maximum number of iterations of the algorithm,
- *repProc* the percentage of bad solutions that will be replaced with new solutions randomly generated,
- *noMut* – number of mutation points for solutions and
- $noIt_{tab}$ **-** the number of iterations for which a mutation of food items in a temporary solution is Tabu.

The algorithm returns a set of optimal or near optimal combinations of food items that will form an older adult daily menu. The generated menus will match the nutritional or dietary constraints as well as the older adult wishes and needs. The algorithm consists of an initialization stage and an iterative stage. In the initialization stage (lines 4-6 of Algorithm 1) the initial population of individuals is generated based on the food service offers available in the marketplace, the older adult's profile and dietary recommendations. Then the iterative stage (lines 7-25 of Algorithm 1) identifies the optimal or near-optimal configuration of food items matching the defined constraints. The operations below are performed in the iterative stage, until a stopping condition (i.e. the predefined number of iterations is reached or a solution with acceptable fitness value is found) is satisfied:

(i) Locally optimal solution is identified among the set of individuals in the population.

(ii) For each individual in the population, the steps below are performed:

a. The optimal number of child seeds is computed using relation (12) and the child seeds are generated.
b. The child seeds are submitted to a mutation process by taking into account the defined constraints (i.e. older adult profile and dietary recommendations and the food service offers), and the information stored in the long-term and short-term memory structures.
c. The long term memory structure is updated.
d. The best child is identified from the population of children.
e. The population of children is updated by applying a path relinking based strategy between the children population and the locally optimal solution.
f. The short term memory structure is updated.

(iii) The updated population of children is added to the population of individuals, and a number *repProc* representing the worst individuals in the whole population are replaced with new individuals randomly generated.

The number of children seeds of a population individual is calculated as:

$$nrSeeds(dailyMenu) = maxSeeds - \frac{(maxSeeds - minSeeds) * Population_{index}}{maxPopSize}\ , \qquad (12)$$

where:

(i) *maxSeeds* **-** represents the maximum number of children seeds,

(ii) *minSeeds* **-** represents the minimum number of seeds,

(iii) *maxPopSize* **-** is the maximum population size, and

(iv) $Population_{index}$ **-** is the index of the plant (daily menu) in the *Population* set (the plants in the *Population* set are stored in a descending order according to their fitness value – the best plant has index 1).

ALGORITHM 1: Hybrid Invasive Weed Optimization

```
Input: foodOffers, personalProfile, dietaryRec, initialPopSize,
       maxSeeds, minSeeds, noIt, noMut, repProc
Output: older adult daily menu
Begin
  Population = RandomlyGeneratePop (initialPopSize, foodOffers, personalProfile, dietaryRec)
  ChildPop = {}
  M_S = {}, M_L = {}
  while (stopping condition not satisfied) do
    sol_opt = GetHighestFitnessSolution (Population)
    foreach dailyMenu in Population do
      noSeeds = ComputeNoSeeds (maxSeeds, minSeeds, GetInd (dailyMenu, Population), maxPopSize)
      ChildSeeds = GenerateChildSeeds (plant, noSeeds)
      ChildSeeds* = Mutate (ChildSeeds, foodOffers, personalProfile, dietaryRec, M_S, M_L, noMut, noIt)
      Update (M_L)
      ChildPop = ChildPop U ChildSeeds*
      bestSeed = GetHighestFitnessSolution(ChildSeeds*)
      if (Fitness(bestSeed) > Fitness(sol_opt)) then
        sol_opt = bestSeed
        ChildSeeds* = ChildSeeds*-{bestSeed}
      end if
    end foreach
    ChildPop = ChildPop U PathRelinking (GetHighestFitnessSolution (ChildSeeds*), sol_opt)
    Update(M_S)
    Population = Population U ChildPop
```

  Population = *ReplaceLowestFitness* (Population, repProc)

end while

return *GetHighestFitness*(Population)

End

3.3. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a bio-inspired optimization meta-heuristic inspired by the foraging behavior of birds in nature. The meta-heuristic uses a set of particles, each particle being defined by a position and a velocity, to search for the global optimum of an NP-hard optimization problem. The particles iteratively update their position according to the individual local optimal position and to the global optimal position encountered so far.

The new position of a particle x_i^{t+1} is formally defined as [7]:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{13}$$

where x_i^t is the current position of the particle, and v_i^{t+1} is the new velocity of the particle.

Relation (13) can be rewritten as (since the time difference between steps is 1):

$$v_i^{t+1} = x_i^{t+1} - x_i^t \tag{14}$$

The new velocity of a particle v_i^{t+1} can be also formally defined as (adapted from [7]):

$$v_i^{t+1} = a * v_i^t + b * \left|x_{best-p}^{t-1} - x_i^t\right| + c * |x_{best-swarm}^{t-1} - x_i^t| \tag{15}$$

where v_i^t is the current velocity of the particle, x_{best-p}^{t-1} is the best position encountered so far by the particle, x_i^t is the current position of the particle, $x_{best-swarm}^{t-1}$ is the best position encountered so far by a particle from the swarm, and a, b and c are constants that weight the importance of each component part of the velocity formula ($a + b + c = 1$).

3.3.1. Diet aware food ordering problem mapping

To map the Particle Swarm Optimization meta-heuristic to the problem of selecting and generating food service combinations that match the older adult preferences and dietary constraints, the following concepts need to be adapted and defined: particle, particle position, particle velocity, addition operator, subtraction operator and multiplication operator.

We define a *particle* as a software agent having a position represented as a candidate combination of food items provided by various food services part of the DIET4Elders marketplace and the particle velocity as mutations of food items which may improve the recommended food items combination.

In our approach, the *particle position* (i.e. candidate solution for combination of food recommendation – sol -) from the swarm at the current step t is represented as:

$$x_i^t = \begin{pmatrix} {FI_b}_i^t \\ {FI_{s1}}_i^t \\ {FI_l}_i^t \\ {FI_{s2}}_i^t \\ {FI_d}_i^t \end{pmatrix} \tag{16}$$

where FI_b represents the set of food items that can be eaten at breakfast, FI_{s1} and FI_{s2} are sets of food items that can be eaten at snacks, FI_l at lunch, and FI_d at dinner, during one day. In consequence the velocity matrix can be written as:

$$x_i^t = \begin{pmatrix} {FoodItem_{b1}}_i^t & \dots & {FoodItem_{bn}}_i^t \\ {FoodItem_{s11}}_i^t & \dots & {FoodItem_{s1n}}_i^t \\ {FoodItem_{l1}}_i^t & \dots & {FoodItem_{ln}}_i^t \\ {FoodItem_{s21}}_i^t & \dots & {FoodItem_{s2n}}_i^t \\ {FoodItem_{d1}}_i^t & \dots & {FoodItem_{dn}}_i^t \end{pmatrix} \tag{17}$$

The particle velocity is a matrix having 5 lines and the number of columns equal with the particle number of columns, each matrix element being 0 or 1.

In what follows we present the interpretations of the arithmetic operators from the new position determination relation (13) considering the new velocity computation relation (15).

The *subtraction between two particles* $x_i^t - x_j^{t+1}$ generated at two consecutive steps is defined as follows:

$$x_i^t - x_j^{t+1} = \begin{pmatrix} FoodItem_{b1}{}_i^t - FoodItem_{b1}{}_j^{t+1} & \dots & FoodItem_{bn}{}_i^t - FoodItem_{bn}{}_j^{t+1} \\ FoodItem_{s11}{}_i^t - FoodItem_{s11}{}_j^{t+1} & \dots & FoodItem_{s1n}{}_i^t - FoodItem_{s1n}{}_j^{t+1} \\ FoodItem_{l1}{}_i^t - FoodItem_{l1}{}_j^{t+1} & \dots & FoodItem_{ln}{}_i^t - FoodItem_{ln}{}_j^{t+1} \\ FoodItem_{s21}{}_i^t - FoodItem_{s21}{}_j^{t+1} & \dots & FoodItem_{s2n}{}_i^t - FoodItem_{s2n}{}_j^{t+1} \\ FoodItem_{s21}{}_i^t - FoodItem_{s21}{}_j^{t+1} & \dots & FoodItem_{dn}{}_i^t - FoodItem_{dn}{}_j^{t+1} \end{pmatrix} \quad (18)$$

Since the particles are generated in consecutive time intervals and the time difference is 1, by subtracting two positions we will obtain a velocity matrix showing how different the particles are evolving (exploration components of the swarm). Thus we define the subtraction between food items as:

$$FoodItem\ {}_i^t - FoodItem\ {}_j^{t+1} = \begin{cases} 0, if\ FoodItem\ {}_i^t = FoodItem\ {}_j^{t+1} \\ 1, otherwise \end{cases} \quad (19)$$

Considering relations (18) and (19) we define two velocities: v_{best-p}, the velocity with which the current particle is moving away from the best particle encountered so far; and $v_{swarm-p}$, the velocity with which the current particle is moving away from the swarm. In consequence relation (15) becomes:

$$v_i^{t+1} = a * v_i^t + b * v_{best-p} + c * v_{swarm-p} \quad (20)$$

The multiplication of a velocity with a constant is defined as:

$$const * v_i = \begin{cases} const, if\ v_i = 1 \\ 0, otherwise \end{cases} \quad (21)$$

Using the above presented relations we can define the operation of adding a particle position with a particle velocity as:

$$x_i^t + v_i^{t+1} = \begin{cases} FoodItem_i, if\ v_i \in [0,0.5) \\ mutate\ (FoodItem_i, FoodItem_k), if v_i \in [0.5,1] \end{cases} \quad (22)$$

where $mutate$ is a function that randomly chooses another food item ($FoodItem_k$) from the list of available food services from the DIET4Elders marketplace to replace a food item from the current solution. By applying this to the relation (13) we can compute the new position of a particle for our problem of selecting and generating food services combinations that match the older adult preferences and dietary constraints.

For evaluating the fitness of a solution we will use the same formula as for the Hybrid Invasive Weed heuristics presented in relation (3) considering the three main aspects of nutrition, quality and delivery.

3.3.2. PSO based food ordering optimization algorithm

The PSO based food ordering optimization algorithm (see Algorithm 2) takes as input the following parameters:

- *dietaryRec* – the older adult recommended diet,
- *foodOffers* **-** the set of food service offers available in the DIET4Elders marketplace,
- *personalProfile* **-** the personal profile of the older adult captured and stored by means of the DIET4Elders Monitoring Ontology,
- *swarmSize* the number of particles of the swarm and finally the *a*, *b*, *c* constants used in the formula for updating a particle's velocity. The output of the algorithm will be a set of food items forming an optimal daily menu in respect with the constraints imposed regarding nutrition, profile and preferences.

The algorithm consists of an initialization and an iterative stage. In the initialization stage, the initial particles of the swarm are randomly generated taking into account the food items provided by the food services from the marketplace, the constraints given by the older adult food preferences and profile as well as the dietary constraints imposed by the nutritionists (line 4). The particle having the best local position and the particle having the best global position are being initialized considering their fitness function values (lines 6-7).

ALGORITHM 2: PSO based Food Ordering Optimization

Input: foodOffers, personalProfile, dietaryRec, swarmSize, a, b, c

Output: optimal older adult daily menu (D_{Menu})

Begin

SWARM = *RandomlyGeneratePraticles* (swarmSize, foodOffers, personalProfile, dietaryRec)

D_{Menu} = null, t=0

x_{best-p} = getLocalBest (SWARM, foodOffers, personalProfile, dietaryRec)

$x_{swarm-best}$= getGlobalBest (SWARM, foodOffers, personalProfile, dietaryRec)

while (stopping condition not satisfied) **do**

for i = 1 **to** swarmSize **do**

$v_i(t+1) = a * v_i(t) + b * \left|x_{best-p} - x_i(t)\right| + c * |x_{swarm-best} - x_i^t|$

$x_i(t+1)$ = GenerateParticles($x_i(t)$; $v_i(t+1)$; foodOffers, personalProfile, dietaryRec)

x_{best-p}= UpdateLocalBest()

$x_{swarm-best}$= UpdateGlobalBest()

end for

t = t+1

end while

D_{Menu} = *GetHighestFitness*(SWARM)

return D_{Menu}

End

In the iterative stage the following operations are performed at each step until a stopping condition (i.e. the predefined number of iterations is reached or a solution with acceptable fitness value is found) is satisfied:

- All the particles part of the swarm are being iterated and new particles are being generated using the relations (13) and (15) and the operators defined for subtracting particle positions, multiplying the velocity with a constant and summing a particle position with velocity.

- New particles with local best position and global best position are identified among the new set of particles generated.

In the end the particle having the highest value of the fitness function is chosen and represents the daily menu optimally fitting the defined constraints.

4. DIET4Elders support services development

In this section we will detail the development of the DIET4Elders support services focusing on providing details about their design and actual implementation.

4.1. DIET4Elders Support Services Design

To design the DIET4Elders support services we have combined and used ICT models and techniques with knowledge regarding the older adult's nutrition process.

From ***ICT and ambient assisted living*** we adopted the MAPE (Monitoring, Analysis, Planning and Execution) architecture of designing context aware adaptable systems [8]. *A context aware adaptive system* must *understand the context* in which it evolves (i.e. understand the context changes) and *automatically decide* on the actions than need to be executed to *adapt its execution to the context changes.*

This architecture defines four main phases for a system:

- Monitoring phase – all data regarding the system execution and the context of its execution are collected by means of physical and logical sensors;
- Analysis phase – the data collected in the monitoring phase is analysed to make assumptions upon and understand the context or situation in which the system evolves;
- Planning phase – decision process which allows the system to plan the execution of adaptation actions to understand the current context/situation;
- Execution phase – actual implementation and execution of decided actions.

From the ***older adult nutrition perspective*** we have used the knowledge and expertise regarding the nutrition care process. The nutrition care process also defines four main phases: nutrition assessment phase, nutrition problem detection phase, nutrition intervention phase and finally the nutrition monitoring phase.

Based on the above presented knowledge we have decided to combine the ICT perspective with the nutrition perspective aiming to define the DIET4Elders system as an older adult nutrition aware system. Thus we have defined four main support services to be provided by our system (see Figure 14): Nutrition Monitoring Service, Nutrition Assessment Service, Nutrition Problem Identification Service and Nutrition Intervention Service.

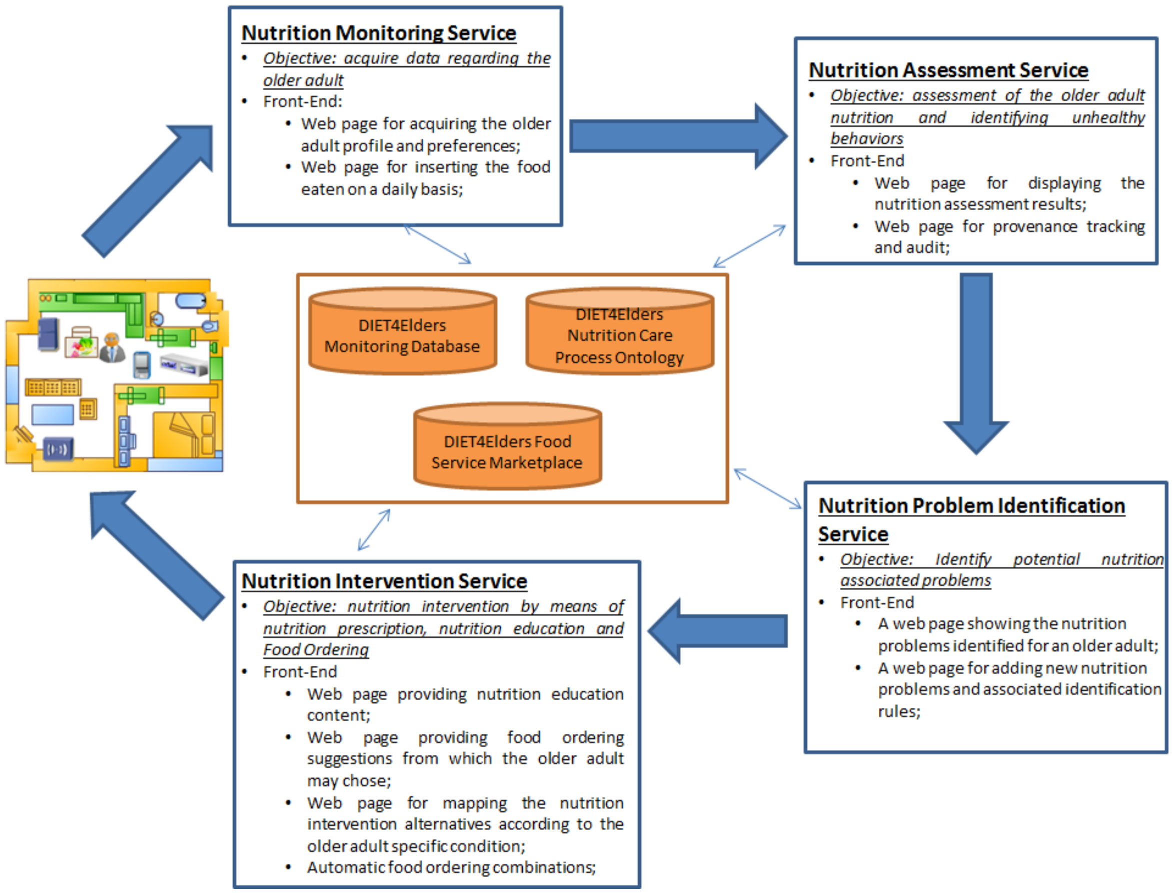

Figure 14. DIET4Elders system service based architecture

4.1.1. Nutrition Monitoring Service

The main objective of the Nutrition Monitoring Service is to get data regarding the degree in which an older adult follows a prescribed diet. In more details it acquires data regarding the older adult nutrition, profile and preferences and stores it into DIET4Elders Monitoring Database.

The data regarding older adult nutrition is gathered by using the TUNSTALL tele-care and tele-health infrastructure described in project deliverable D2.1 - First version of the monitoring infrastructure and data representation. The collected data will be stored into the DIET4Elders Monitoring Database by using the REST API provided by the TUNSTALL tele-care and tele-health infrastructure.

The data regarding the older adult profile and preferences will be gathered by means of the DIET4Elders older adult centric web page. The web page is driven by the Nutrition Monitoring Ontology for acquiring the older adult profile and preferences (the former Older Adult Daily Life Context Model from the project DoW (Description of Work) which is described in deliverable D3.1 - First version of the older adults dietary and self-feeding behaviour assessment techniques) and can be used by the older adult alone or with the help of the associated carer.

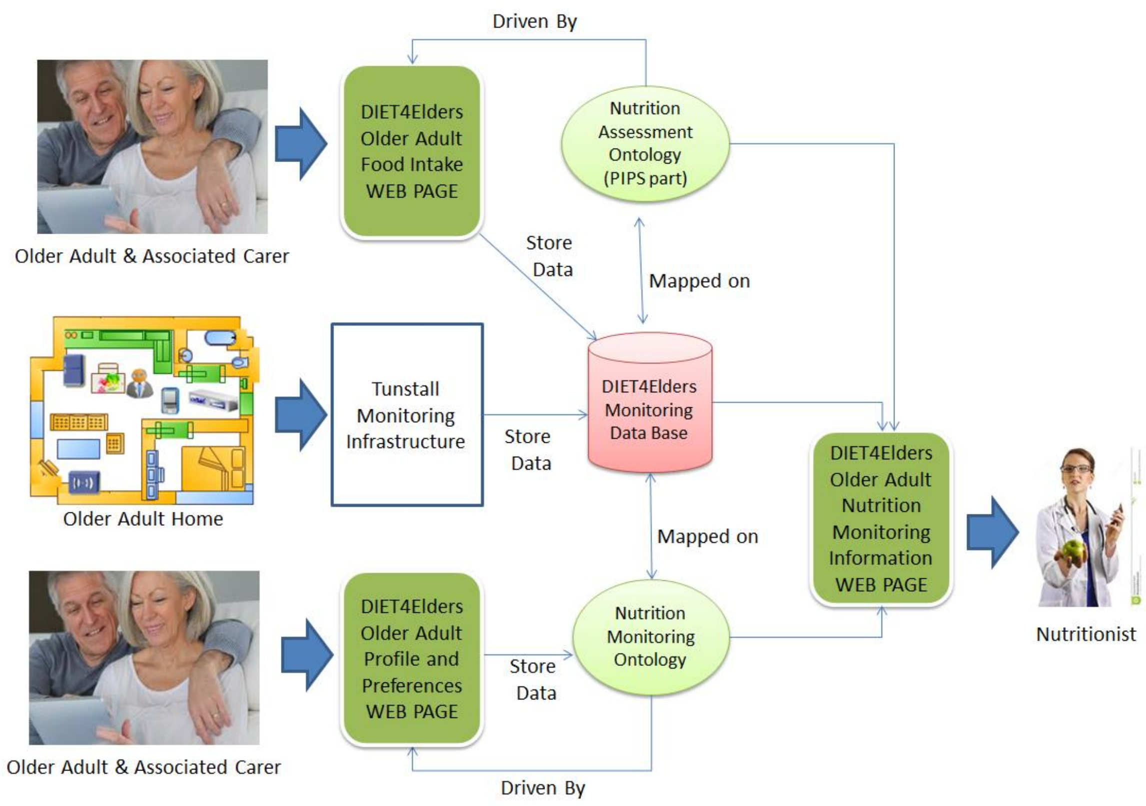

Figure 15. Design of Nutrition Monitoring Service

Figure 15 shows the design of DIET4Elders Nutrition Monitoring Service. The service will be constructed around the DIET4Elders Monitoring Ontology, which will drive the reasoning processes and the content to be displayed dynamically to older adults and to the nutritionist. The ontology will be mapped onto the DIET4Elders Monitoring Database which will contain real time data about the older adult, collected by means of TUNSTALL monitoring infrastructure. The advantage of such approach is the uniform and semantic representation of the monitored data as well as the usage of reasoning techniques for evaluating the older adult nutrition related behaviour.

The Nutrition Monitoring Service will provide graphical user interfaces both to the older adult (or carer) and associated nutritionist. The DIET4Elders Older Adult Food Intake Web Page will guide the older adult to select the meals he/she has eaten during a day. The DIET4Elders Older Adult Profile and Preferences Web Page will allow the older adult to provide information on their profile wishes and needs. The DIET4Elders Older Adult Nutrition Monitoring Information Web Page will display nutritional information of an older adult to its associated nutritionist.

4.1.2. Nutrition Assessment Service

The main objective of the Nutrition Assessment Service is to evaluate the older adult nutrition (based on the monitored data) with the goal of identifying unhealthy older adult behaviours. It is based on the Nutrition Problem Identification Ontology, Nutrition Assessment Ontology and Nutrition Monitoring Ontology, which will drive the older adult nutrition assessment processes.

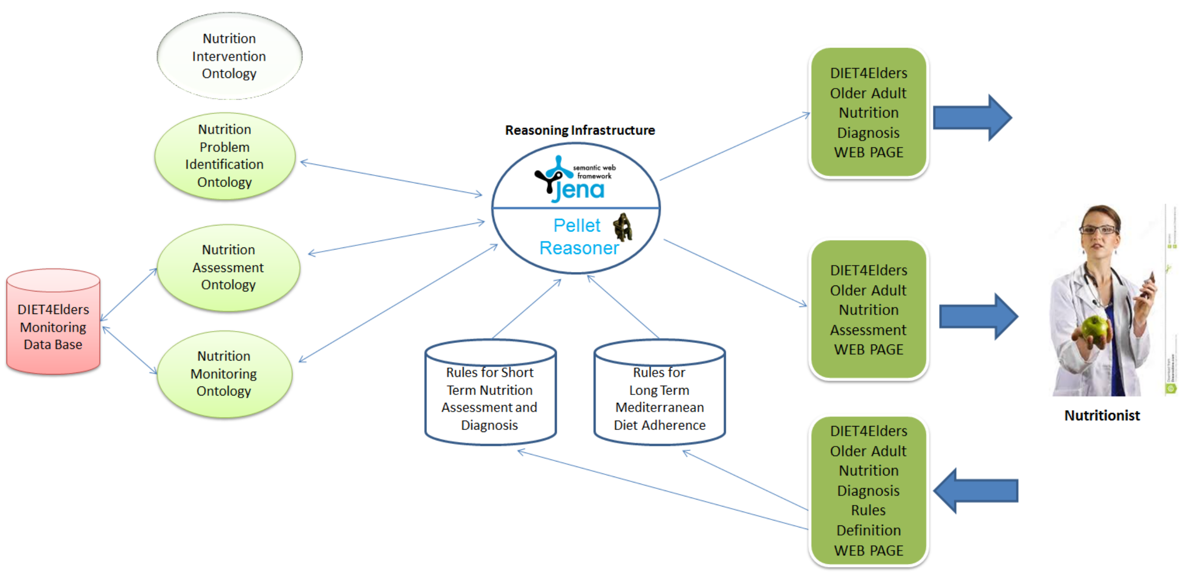

Figure 16. Nutrition assessment service design

The design of the DIET4Elders Nutrition Assessment Service can be seen in Figure 16. Based on the nutrition related knowledge provided in deliverable D3.1 we have designed a reasoning infrastructure capable of detecting two types of older adult un-healthy behaviour: short term un-healthy nutrition assessment (i.e. older adult diet related constraints) and long term un-healthy nutrition assessment and diagnose (i.e. older adult long term adherence to the Mediterranean diet). Details about the techniques for detecting the un-healthy nutrition related behaviours are presented in deliverable D3.1.

The service will provide the following web interfaces for the associated nutritionist:

(i) the DIET4Elders Older Adult Nutrition Diagnosis Web Page will display the short term nutrition diagnosis processes,

(ii) the DIET4Elders Older Adult Nutrition Assessment Web Page will display the long term nutrition assessment processes and

(iii) the DIET4Elders Older Adult Nutrition Diagnosis Rule Definition Web Page will allow the nutritionist to add new nutrition related problems and their associated symptoms to be represented as reasoning rules.

4.1.3. Nutrition Problem Identification Service

The main objective of this service is to assess potential nutrition related problems and associated symptoms based on the monitored data. This service is implemented by defining rules that allow the early identification of older adult symptoms that are associated with nutrition related problems or diseases such as obesity, un-balanced energy intake, malnutrition, etc. The potential nutrition related problems and symptoms are being described by the Nutrition Problem Identification Ontology. This service enacts the nutritionist with the possibility of proactive detection of older adults' nutrition problems thus being able to define nutrition intervention schemes that allow for problem prevention before its actual instauration.

Related to the service web interface, it will provide the following graphical interfaces to the nutritionists: the DIET4Elders Older Adult Nutrition Problem web page showing the diseases identified for an older adult, and the DIET4Elders Older Adult Nutrition Problem Rules Definition web page for adding new diseases and associated identification rules.

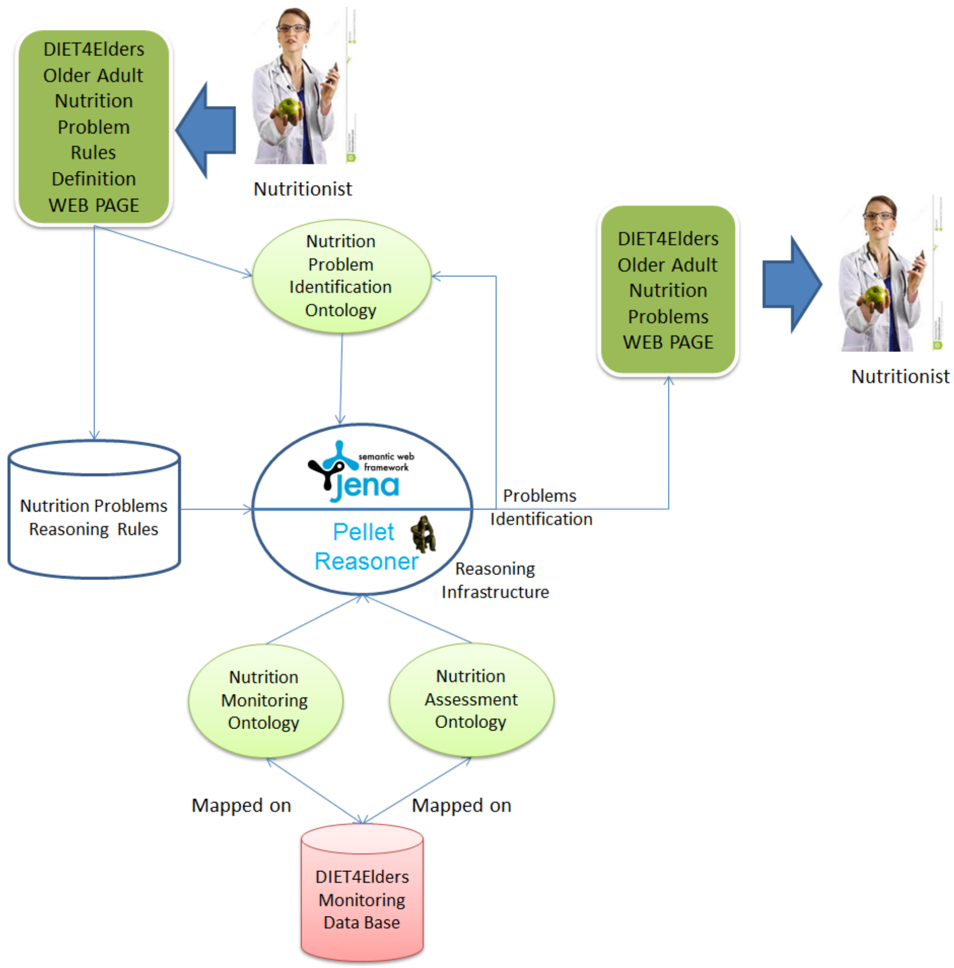

Figure 17. Nutrition Problem Identification Service Design

4.1.4. Nutrition Intervention Service

The nutrition intervention in DIET4Elders is carried out on three major directions (see Figure 18):

(1) Nutrition education - the DIET4Elder Older Adult Nutrition Education Web Page will provide personalized information on nutrition diseases,

(2) Dietary recommendations - the DIET4Elders Older Adult Nutrition Intervention Mapping Web Page will allow the nutritionist to define nutrition plans or diets personalized to the older adult condition and

(3) Diet aware food ordering service - the DIET4Elders Older Adult Food Ordering Web Page will guide the older adults in the process of ordering food which match their nutritional recommendation.

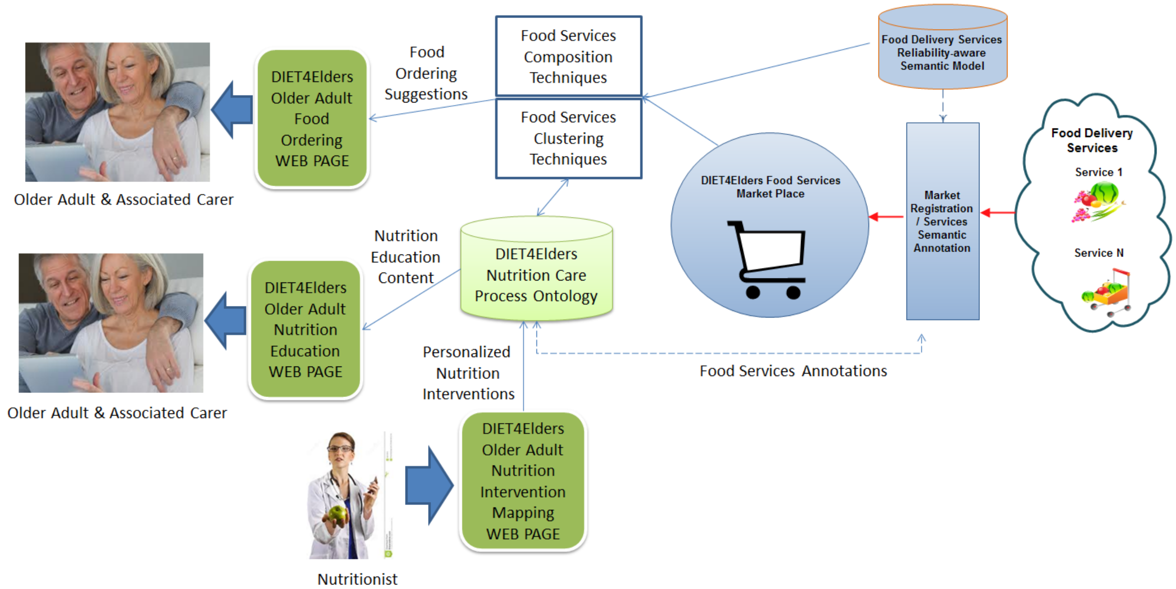

Figure 18. Design of Nutrition Intervention Service

The most innovative aspect of nutrition intervention in DIET4Elders is the diet aware food ordering service.

The first design of the DIET4Elders diet aware food ordering service can be seen in Figure 18. The service aggregates the food delivery services of different providers and will guide the older adults and their informal carers to order the right food to comply with the nutritionists recommended diet. It is centred on the DIET4Elders Nutrition Care Process Ontology and has the DIET4Elders Food Services Marketplace as a fundamental component to define food services in a diet and nutritionally rich context, where food providers can register their services. All registered and market available food services will conform to a service description specification using concepts from the Food Provider Ontology and Reliability Monitoring Ontology (see Chapter 2 for description of these ontologies). The service provides the selection and composition of food items supplied by various food services registered within the marketplace that optimally match the older adult wishes and preferences as well as his/her dietary constraints. This functionality is provided by means of the hybrid bio-inspired heuristics presented in Chapter 3.

4.2. DIET4Elders Support Services Implementation

In this section we will present details about the support services implementation process. We have started by defining use cases based on the DIET4Elders scenarios aiming to identify the end-users interactions and functional requirements that need to be implemented through the DIET4Elders support services web interfaces (section 4.2.1). In section 4.2.2, it is showed how the functional requirements are actually being implemented through the web interfaces of support services.

4.2.1. DIET4Elders System Use Cases

Based on the general architecture organization and functionality specified in Section 4.1, the main use cases have been defined and used to identify the interaction between the users and the support services that need to be implemented.

A *use case* describes the interaction between an individual and the system. Each use case describes an event that may occur for a short period of time, but may consist of intricate details and interactions between the actor and DIET4Elders support service web interface. The use cases are used to inform developers on the logical steps that need to be considered during the development cycle.

The use cases are described using a common table schema (see Tables 6 to 16 below). The main section is the Main Flow, where the use case is broken down into an ordered list of interactions between the end-user and the system. The use cases are not fully complete with respect to the possible functionality and client type permutations, but enough to cover all forms of support services web interfaces' interactions. The use case actors are the DIET4Elders end-users: Older Adult, Informal carer, Nutritionist and Food Provider.

Table 6. Older adult profile and preferences

Use case number	1
Use case Name	Older adult profile and preferences
Actors	Primary user (Older Adult, Informal Carer)

Summary	Covers all the steps to configure and edit older adults profile and preferences
trigger/intent	*The user* enters the DIET4Elders front-end profile/preferences screen
Preconditions	*Older adult* is registered and using the system and wishes to update/modify their profile or preferences
Flow of events: (Main flow)	1. *Older adult* clicks on profile/preferences link 2. The user info is retrieved from the ontology and showed to the user 3. *Older adult* changes their profile/preferences data 4. When *older adult* presses the save button on the web, the data is stored in the ontology, replacing old user data
Alternate flows	• request data does not meet the correct format • *older adult* abandons the frontend
Exceptional flows	Data can't be saved on the ontology. system shows an error message
Displayed information	Form to replace the actual user information
Post-conditions	*Older adult* has its profile/preferences updated
Relation to other use cases	Nutrition Assessment, Nutrition Problems Assessment and Food ordering depend on this information

Table 7. Older adult food intake

Use case number	2
Use case Name	Older adult food intake
Actors	Primary user (Older Adult, Informal Carer)
Summary	*Older adult* registers the food intake in the DIET4Elders system. Covers all the steps to register the food intake
trigger/intent	*Older adult* clicks on the link *Register intake*
Preconditions	*Older adult* is on the DIET4Elders system and wants to register its food intake
Flow of events: (Main flow)	1. *Older adult* selects from the menu the type of food to register 2. *Older adult* selects the amount of food 3. Repeats the process all the times needed 4. *Older adult* clicks on save 5. The food intake is registered in the ontology

Alternate flows	• request data does not meet the correct format • *Older adult* abandons the frontend
Exceptional flows	Data can't be saved on the ontology, shows an error message
Displayed information	Food intake has been saved successfully
Post-conditions	The new food intake provided by the *older adult* is stored on the database
Relation to other use cases	Nutrition Assessment and Nutrition Problems Assessment depend on this information

Table 8. Nutrition Monitoring Information

Use case number	3
Use case Name	Nutrition Monitoring Information
Actors	Nutritionist
Summary	The system monitors older adult data relevant for assessing their feeding behaviour. The *Nutritionist* can check the monitored data on the website
trigger/intent	*Nutritionist* clicks on the *user monitoring* web page
Preconditions	*Nutritionist* has registered an *older adult* as a person on its care and wants to monitor its food intake
Flow of events: (Main flow)	1. *Nutritionist* selects from the menu *monitor user* link 2. From the *database* all users linked with the *Nutritionist* are returned 3. *Nutritionist* selects an older adult 4. All the monitored information for the selected older adult is returned
Alternate flows	-
Exceptional flows	• Data can't be loaded from the ontology. system shows an error message • No information displayed due to sensors related technical problems. The older adult needs to be contacted.
Displayed information	Older adult monitored data
Post-conditions	The nutritionist knows the adult food intake behaviour
Relation to other use cases	Nutrition Assessment and Nutrition Problems Assessment depend on this information

Table 9. Nutrition Assessment

Use case number	4
Use case Name	Nutrition Assessment
Actors	Nutritionist, Older Adult, Informal Carer
Summary	*The system* can make an assessment of the older adult nutrition process and identifies unhealthy behaviours.
trigger/intent	*Nutritionist* clicks on the *Nutrition* assessment link
Preconditions	*Nutritionist* has registered an *older adult* as a person on its care and wants to assess its nutrition to detect unhealthy behaviours
Flow of events: (Main flow)	*1. Nutritionist* selects from the menu *monitor user* link 2. From the *database* all users linked with the *Nutritionist* are returned 3. *Nutritionist* selects an older adult 4. The identified unhealthy behaviours for the selected older adult are returned
Alternate flows	-
Exceptional flows	-
Displayed information	Assessed unhealthy behaviour of older adult
Post-conditions	The nutritionist is informed about the inferred unhealthy behaviour
Relation to other use cases	Older adult profile and preferences, adult food intake and Nutrition Monitored information

Table 10. Nutrition Care Process Audit

Use case number	5
Use case Name	Nutrition Care Process Audit
Actors	Nutritionist
Summary	The system may audit the nutrition care process using a provenance tracking approach. The audit data is displayed to the nutritionist.
trigger/intent	*Nutritionist* clicks on the *Nutrition audit* link
Preconditions	*Nutritionist* has registered an *older adult* as a person on its care and wants to audit the system implemented nutrition care process

Flow of events: (Main flow)	*1. Nutritionist* selects from the menu *audit* link 2. From the *database* all users linked with the *Nutritionist* are returned 3. *Nutritionist* selects an older adult 4. The audit information for the nutrition care process associated to the selected older adult is returned
Alternate flows	-
Exceptional flows	• Data can't be loaded from the ontology. system shows an error message • Data can't be saved in the ontology
Displayed information	Data regarding the nutrition care process audit
Post-conditions	The nutritionist is informed about the accuracy of the nutrition care process implemented by the DIET4Elders system.
Relation to other use cases	ALL the other use cases.

Table 11. Nutrition Problems Rules Definition

Use case number	6
Use case Name	Nutrition Problems Rules Definition
Actors	Nutritionist
Summary	Through the website, the nutritionist can define rules that permit identify potential nutrition associated problems
trigger/intent	*Nutritionist* clicks on the *Define rules* link
Preconditions	*Nutritionist* is registered as a user of the DIET4Elders system and wants to create rules to identify potential nutrition associated diseases
Flow of events: (Main flow)	1. *Nutritionist* clicks on the define rules link 2. *Nutritionist* uses the monitoring database, the nutrition monitoring ontology and the nutrition assessment ontology to create rules to identify potential nutrition associated diseases 3. The new rule is created
Alternate flows	-
Exceptional flows	-
Displayed	Form to create the rules

information	
Post-conditions	the new rules have been created
Relation to other use cases	-

Table 12. Nutrition Problems Assessment

Use case number	7
Use case Name	Nutrition Problems Assessment
Actors	Nutritionist, Older Adult, Informal Carer
Summary	*The system* can make an assessment of the older adult nutrition process and can identify potential nutritional problems.
trigger/intent	*Nutritionist* clicks on the *Nutrition problems assessment* link
Preconditions	*Nutritionist* has registered a *older adult* as a person on its care and wants to assess its nutrition to detect potential nutritional problems
Flow of events: (Main flow)	1. *Nutritionist* selects from the menu *monitor user* link 2. From the *database* all users linked with the *Nutritionist* are returned 3. *Nutritionist* selects an older adult 4. The identified nutritional problems for the selected older adult are returned
Alternate flows	-
Exceptional flows	-
Displayed information	Assessed nutritional problems of older adult
Post-conditions	The nutritionist is informed about the inferred nutritional problems
Relation to other use cases	Older adult profile and preferences, adult food intake and Nutrition Monitored information

Table 13. Nutrition Prescription

Use case number	8
Use case Name	Nutrition Prescription

Actors	Nutritionist
Summary	Through the website, the nutritionist can define nutrition prescription or diets that must be followed by an older adult
trigger/intent	*Nutritionist* clicks on the *Nutrition Prescription* link
Preconditions	*Nutritionist* is registered as a user of the DIET4Elders system and wants to create a nutrition prescription and to associate it to an older adult
Flow of events: (Main flow)	1. *Nutritionist* clicks on the nutrition prescription link 2. *Nutritionist* creates a new nutrition prescription 3. The nutrition prescription is associated to a specific older adult 4. The new prescription is saved in the nutrition intervention ontology and used to drive the food ordering process for that older adult
Alternate flows	-
Exceptional flows	-
Displayed information	Form to create the prescription
Post-conditions	The new prescription has been created, associated to the older adult and used during food ordering
Relation to other use cases	-

Table 14. Nutrition Education

Use case number	9
Use case Name	Nutrition Education
Actors	Older Adult
Summary	Through the web site the *older adult* can see some material related with her/his food intake unhealthy behaviour suggested by the system
trigger/intent	*Older Adult* clicks on the *Nutrition Education* link
Preconditions	*Older Adult* is registered on the system and wants to get some advice
Flow of events: (Main flow)	1. *Older adult* clicks on Nutrition education link/tab 2. Based on the older adult's profile, specific information is presented
Alternate flows	-
Exceptional flows	Data can't be loaded from the ontology. system shows an error message

Displayed information	Education material related with the user diseases
Post-conditions	*Older adult* can see education material related with its diseases
Relation to other use cases	-

Table 15. Diet Aware Food Suggestions

Use case number	10
Use case Name	Diet Aware Food Suggestions
Actors	Older Adult, Informal Carer
Summary	Through the website the older adult will receive food ordering suggestions (one meal) according to their preferences and nutrition diseases. *Older adult user* can order food by selecting from different menus showing food prescribed for their diseases.
trigger/intent	*Older adult* clicks on the *food ordering* link/tab
Preconditions	*Older adult* is registered in the system *Older adult* wants to order something to eat
Flow of events: (Main flow)	1. *Older adult* clicks on the *food ordering* link/tab 2. *Older adult* specifies criteria reflecting their current food preferences (e.g. desired ingredients) 3. *Food ordering* suggestions are displayed according to the *older adult*'s recommended diet and preferences 4. *Older adult* navigates through the suggested food to construct a food package
Alternate flows	-
Exceptional flows	Data can't be loaded from the ontology. System shows an error message
Displayed information	Food satisfying the user's nutrition prescription and preferences
Postconditions	*Older adult* has selected and ordered food
Relation to other use cases	-

Table 16. Automatic Food Ordering Combinations

Use case number	11
Use case Name	Automatic Food Ordering Combinations
Actors	Older Adult, Informal Carer
Summary	Through the website the older adult will receive food combination suggestions (for a longer time period) according to their preferences and nutrition diseases. *Older adult user* can order food by selecting from these different combinations.
trigger/intent	*Older adult* clicks on the *automatic food ordering* link/tab
Preconditions	*Older adult* is registered in the system *Older adult* wants to order something to eat
Flow of events: (Main flow)	1. *Older adult* clicks on the *automatic food ordering* link/tab 2. *Automatic food ordering* generates food combinations matching the user's diet constraints and preferences 3. *Older adult* navigates through the suggested combinations to select the one he/she likes
Alternate flows	-
Exceptional flows	Data can't be loaded from the ontology, system shows an error message
Displayed information	Combinations of food associated with the users nutrition prescription and preferences
Post-conditions	*Older adult* has selected and ordered a food combination
Relation to other use cases	-

4.2.2. Web Interface Implementation

DIET4Elders web interfaces have been developed in an iterative process that starts with the creation of mock-up interfaces based on the above presented use cases. These mockups are validated by a committee of experts and end users, and as result of this validation, the mock-ups are modified. Once mock-ups are accepted, the development of the user interfaces starts.

The development of the user interfaces in Diet4elders has been done using the latest technologies like HTML5 (HyperText Markup Language)[4] or Bootstrap[5]. HTML5 is the last version of the well-known HyperText Markup Language, the standard markup language used to create web pages. Bootstrap is a free collection of tools for creating websites and web applications. It contains HTML and CSS-based design templates for text, forms, buttons, navigation and other components. It also contains optional JavaScript[6] extensions. Bootstrap is open source and available on GitHub[7].

Web Interfaces are the means used by the users to interact with the DIET4Elders system, so user interfaces will give access to the support services (see Figure 19).

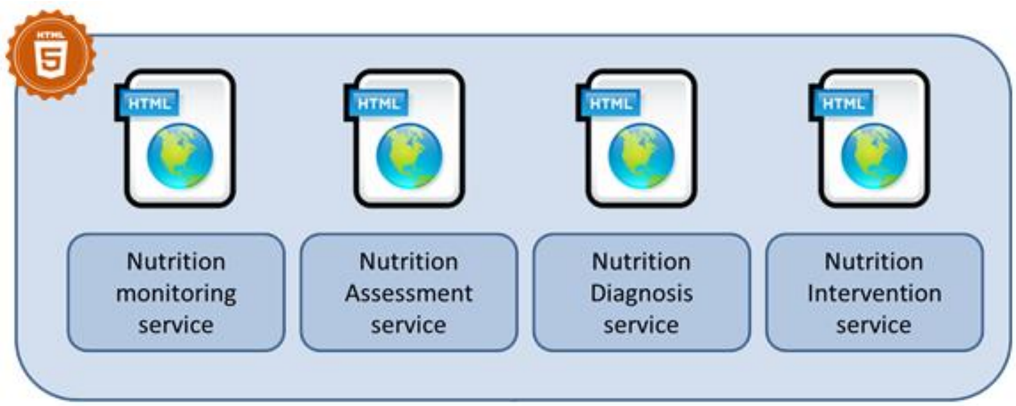

Figure 19. DIET4Elders Web Interface Implementation

In the following sub sections we present the end-user web interfaces of the DIET4Elders system grouped depending on the end user (actor) that is going to use the interfaces.

4.2.2.1. Older Adult Web Interfaces

These interfaces have been developed using certain design guidelines elaborated to comply with the requirements of the older adults.

[4] http://www.w3.org/TR/2014/REC-html5-20141028/
[5] http://getbootstrap.com/
[6] http://en.wikipedia.org/wiki/JavaScript
[7] https://github.com/

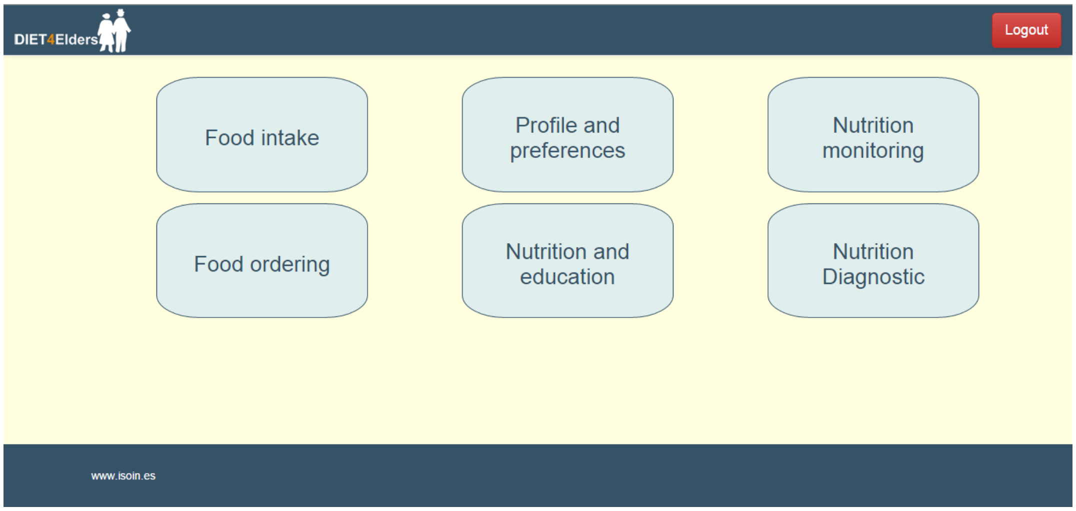

Figure 20. Older adult main menu web interface

In Figure 20, the main menu for the older adult is shown. Using this menu the older adult can select one of the following options:

- **Food intake**: In this option the user can select the food he/she is eating daily. Entering in the food intake option, the older adult can select from a varied list of food items (see Figure 21).

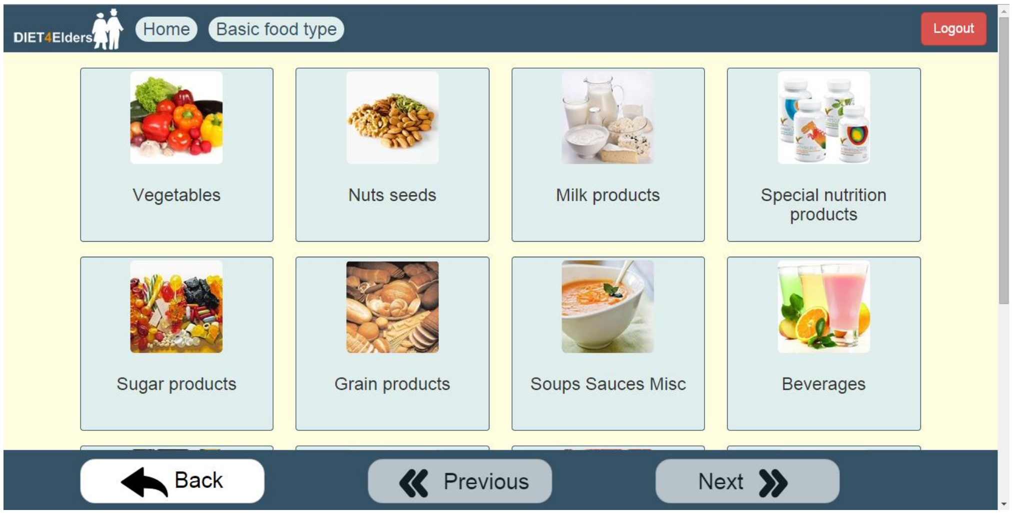

Figure 21. Food intake menu web interface

- **Profile and preferences**: In this option, the user can insert the information about their profile, diseases, preferences and other useful clinical information (Figure 22).

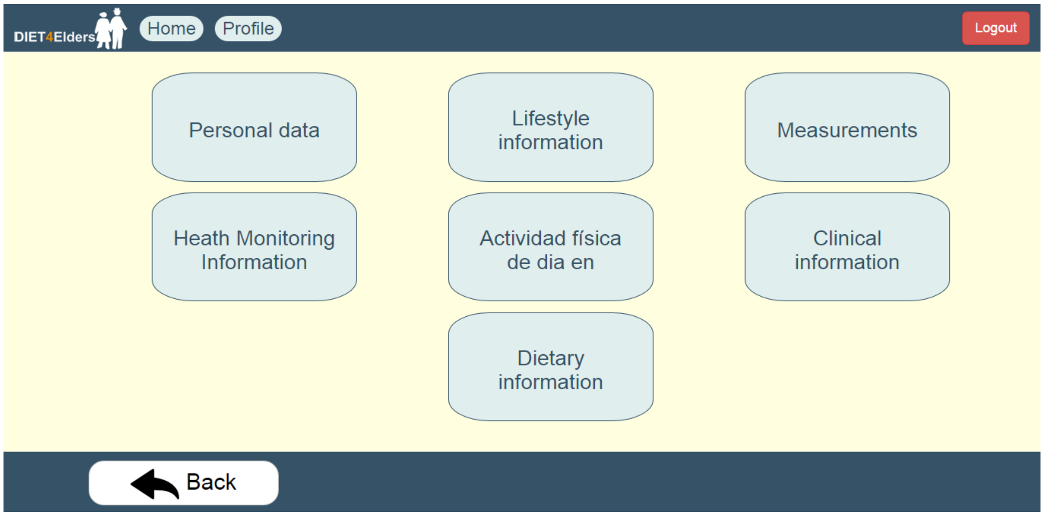

Figure 22. Profile and preferences web interface

Once the user has selected the profile menu a secondary menu is shown, where the user selects which kind of information he/she is going to introduce. By selecting for example personal data, the user can insert their personal information in the system; this information is immediately sent to the career or nutritionist (Figure 23).

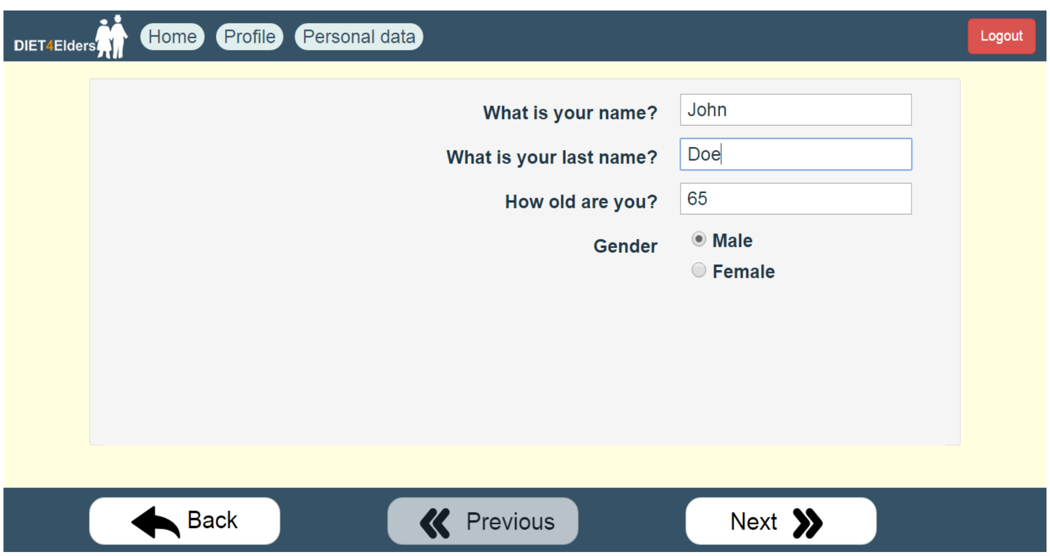

Figure 23. Older adult information (Personal data) web interface

- **Nutrition monitoring**: The user can see the values of the different sensors linked to the DIET4Elders system, and also can see a summary of the food eaten and nutritional information about this food (see Figure 24).

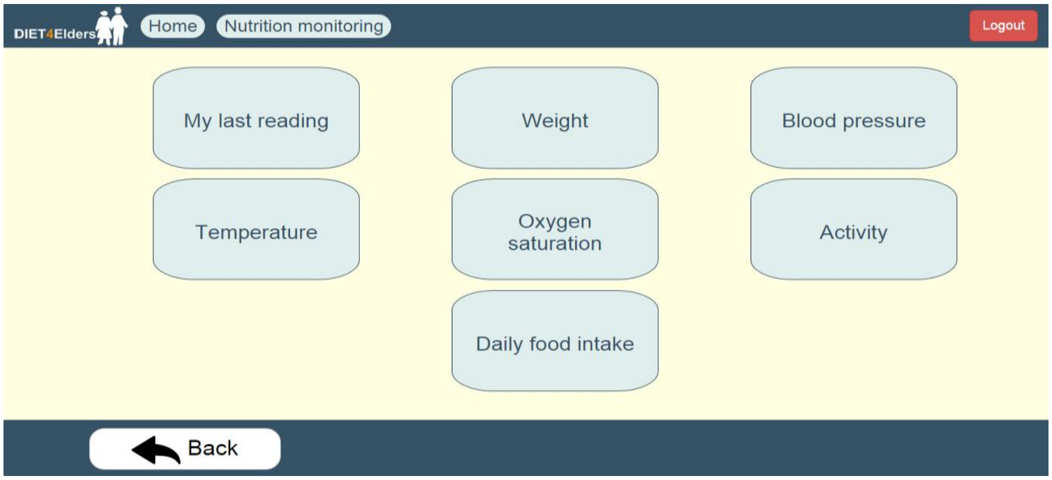

Figure 24. Nutrition monitoring submenu

- **My Last Reading**: The user can see the last values gathered from the sensor, such as weight, blood pressure, temperature, etc. (see Figure 25).

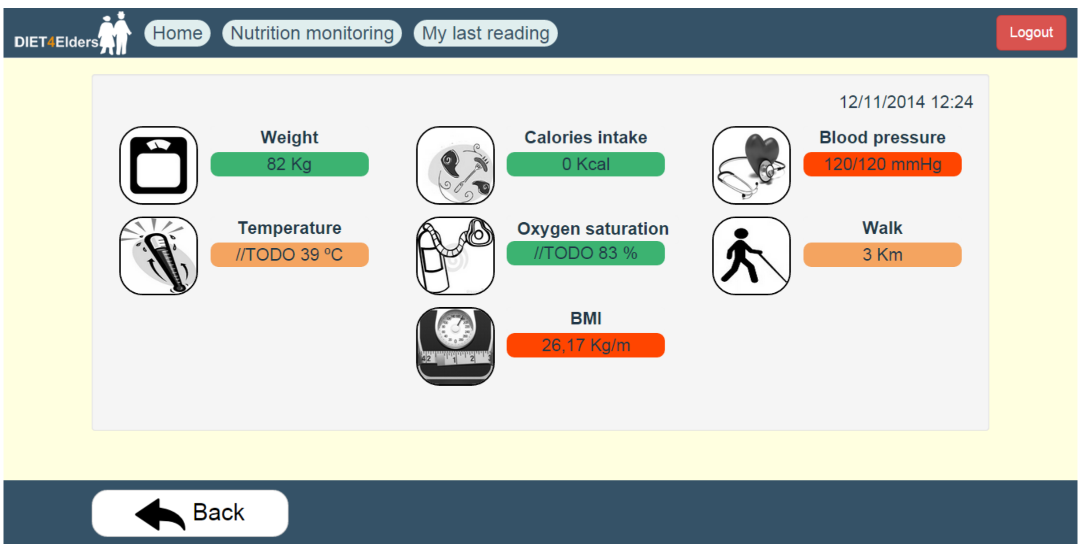

Figure 25. Older adult last reading web interface

- **Daily food intake**: Shows nutritional information about the food daily eaten as illustrated in Figure 26.

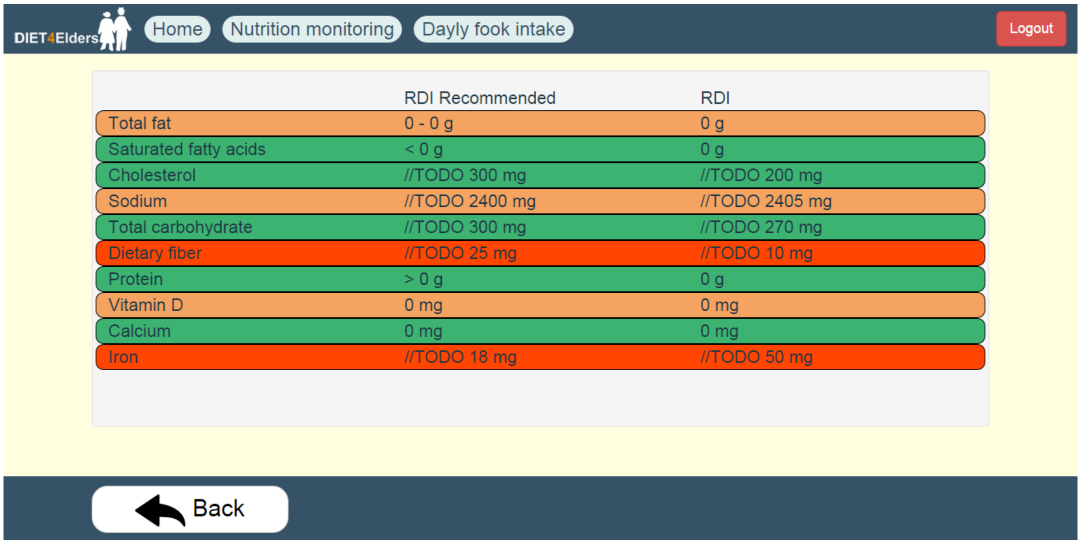

Figure 26. Daily food intake web interface

- **Food ordering**: the user can select and order food for a day of the week, selecting also if it's for breakfast, lunch or dinner (see Figure 27).

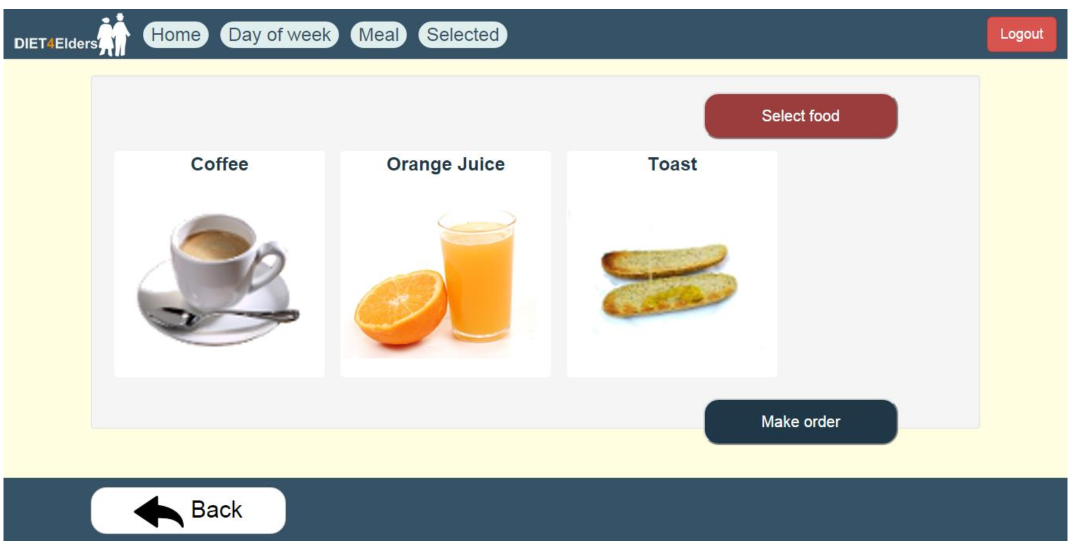

Figure 27. Food ordering web interface

- **Nutrition and education**: the user can see multimedia information to learn about nutrition and nutrition related diseases (Figure 28).

Figure 28. Nutrition and education web interface

4.2.2.2. Nutritionist Web Interfaces

These user interfaces do not follow the same guidelines used for the older adults as they do not have same necessities, and have been developed using clean and minimalist style trying to simulate a clinical environment where nutritionists feel comfortable.

Once the nutritionist is logged on the system, the system shows the work environment's main menu (see Figure 29).

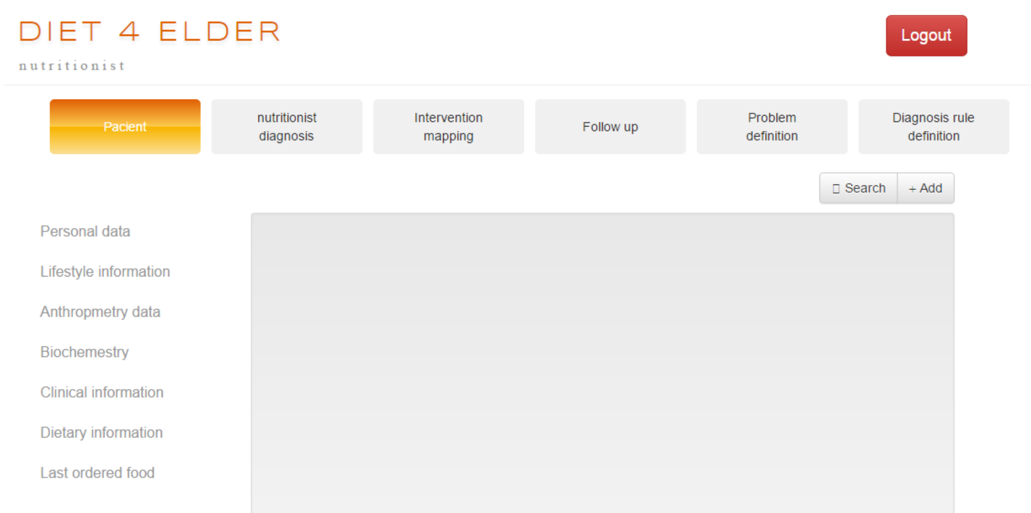

Figure 29. Nutritionist work environment with main menu

By clicking on the add button the nutritionist can select an older adult to be linked with him. Once the nutritionist has selected an older adult he/she can have access to specific information. By clicking on the buttons of the menu, the nutritionist can visualize different information. The same can be done by clicking on the search button if the user is already linked with the nutritionist. Clicking on this button will show the list of patients for the nutritionist, where he can select one.

The DIET4elders system provides the following main web interface tabs to nutritionists:

- **Patient tab:** the nutritionist can have access to the information entered by the older adult and collected by the sensors of the system (see Figure 30), such as personal data, lifestyle information, anthropometric data, clinical information, dietary information, etc.

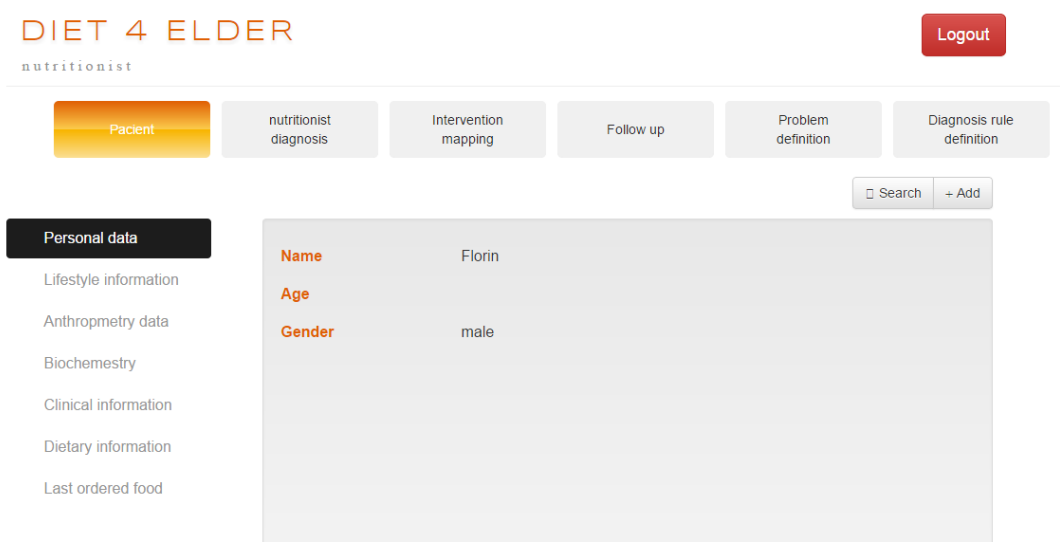

Figure 30. Patient tab web interface

- **Intervention mapping tab:** the nutritionists can establish dietary goals, goals for influencing behavior and intervention options (Figure 31).

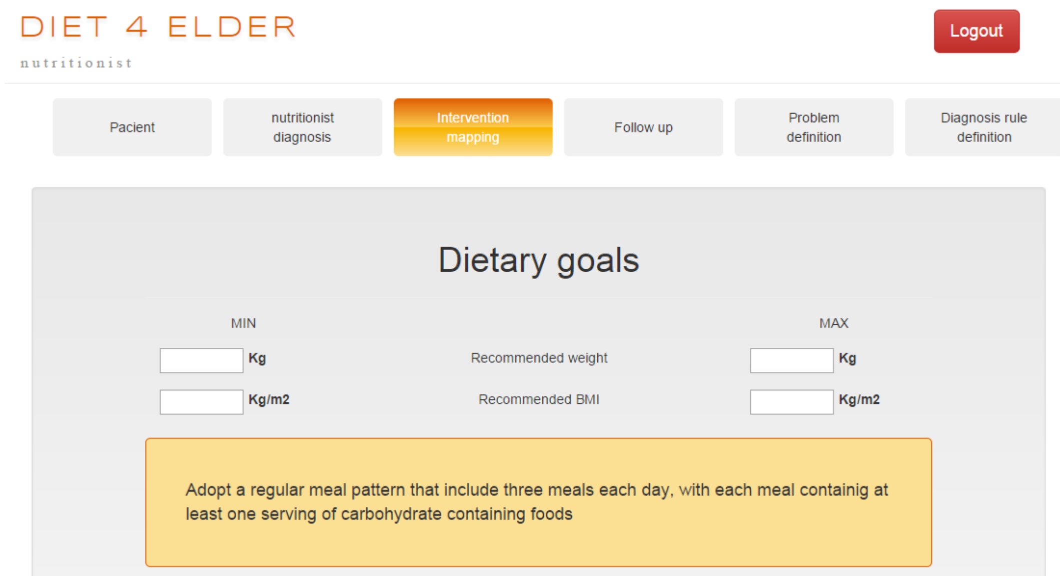

Figure 31. Intervention mappings (Dietary goals) web interface

- **Follow up tab:** the nutritionists can see the nutritional information collected from user behavior feeding, with the colors showing if the defined limits have been broken (see Figure 32).

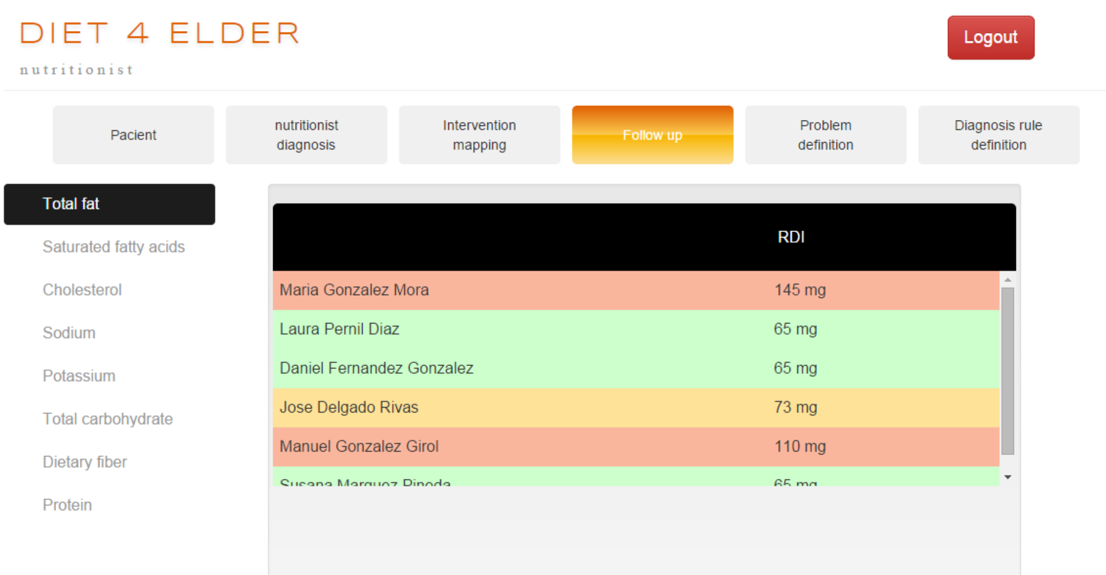

Figure 32. Nutrition care process follow up tab

4.2.2.3. Food Provider Web Interfaces

Web interfaces for food providers are created in the same way as for nutritionists. Obviously necessities for this kind of users are different so the menu changes to give the user the appropriate functionality.

For the food providers the main menu gives functionality for:

- **Contact details**: Give details about the company (see Figure 33)
- **Access policy**: Restrictions on the provider's service availability (e.g. available to everyone, to a particular age group, etc).
- **Quality and safety standards**: List of standards that the provider is compliant with (Figure 34).
- **Geographical coverage**: Geographical area of coverage of the provider.
- **Menu choices**: Give the possibility to upload menus to the system (Figure 35).
- **Supported service options**: Give some optional services (e.g. professional help on food delivery).

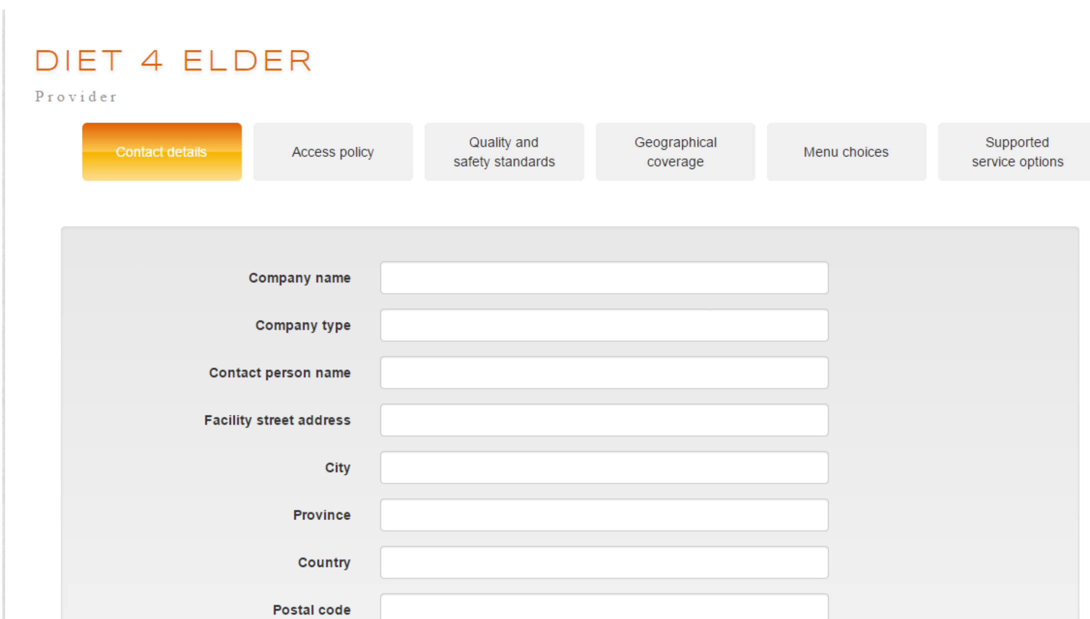

Figure 33. Food provider menu and contact details web interface

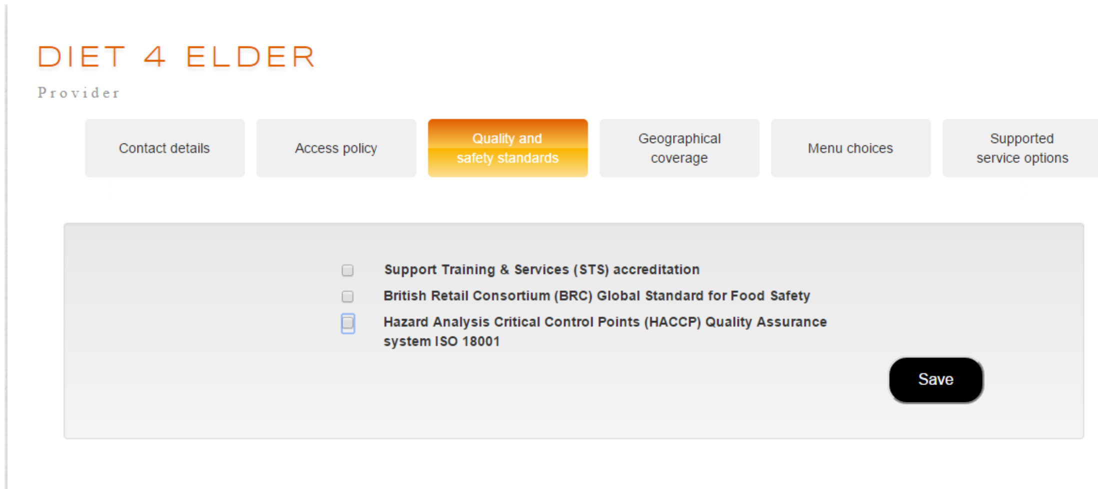

Figure 34. Quality and safety standards web interface

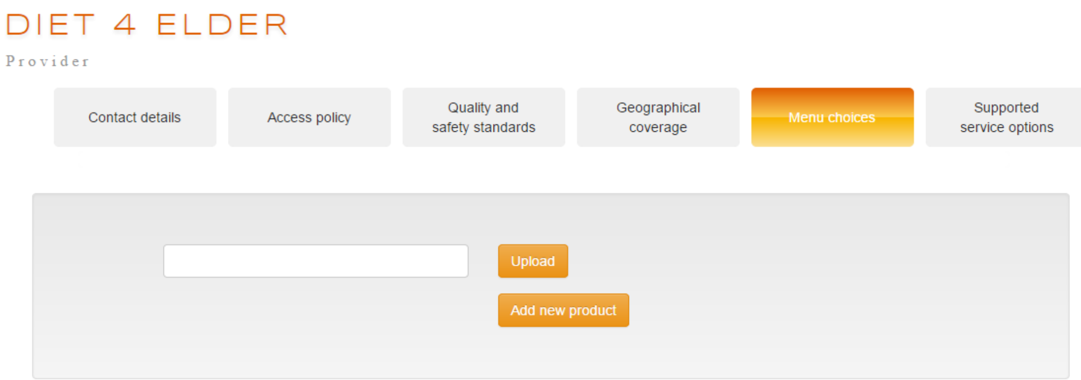

Figure 35. Adding new menu choices web interface

5. Conclusions

In this deliverable we have presented the achievements obtained in the tasks corresponding to the WP4 - Self-feeding support services development, during the reported period (from the start of the WP4 in month 6 until month 18).

In the *Task 4.1: Food delivery services reliability-aware semantic model* the main achievement reported is the development of the DIET4Elders reliability-aware marketplace for food delivery services. The marketplace is based on rich semantic models for capturing the nutritional aspects and various other food-related characteristics as well as reliability dimensions of the registered services.

In the *Task 4.2: Development of hybrid bio-inspired techniques for food services dynamic discovery and composition* the main achievement reported is the definition of two hybrid bio-inspired techniques which are used to enact the older adult diet aware food ordering: Hybrid Invasive Weed and Particle Swarm Optimization.

In the *Task 4.3: DIET4Elders support services development* the main achievement presented in this deliverable is the design and implementation of the four DIET4Elders support services: Nutrition Monitoring Service, Nutrition Assessment Service, Nutrition Problem Identification Service and Nutrition Intervention Service.

As future work we plan to integrate all the above presented models, techniques and services into the DIET4Elders system first prototype which will be evaluated with the help of the older adults and the evaluation. The evaluation results will be used to refine and improve models, techniques and services and if necessary, the new ones will be developed.

6. References

[1] Cornel Christian Sieber, Malnutrition and appropriate nutritional care, Nutrition Day Conference 2010, www.european-nutrition.org.

[2] Consumer Focus Scotland: Food services for older people in the community. Final Report, 2011.

[3] L. Barakat, S. Miles, I. Poernomo, M. Luck: Efficient multi-granularity service composition. In: 2011 IEEE International Conference on Web Services. 2011, pp. 227-234.

[4] A. R. Mehrabian and C. Lucas, A novel numerical optimization algo-rithm inspired from weed colonization, Ecol. Inform. , vol. 1, no. 4, pp. 355–366, 2006.

[5] F. Glover and M. Laguna,Tabu Search, Kluwer Academic Publishers, Norwell, MA, 1997.

[6] F. Glover, M. Laguna, Scatter Search and Path Relinking: Foundations and Advanced Designs, New Optimization Techniques in Engineering Studies in Fuzziness and Soft Computing, vol. 141, pp 87-99, 2004.

[7] J. Kennedy, and R. Eberhart, Particle Swarm Optimization, Proceedings of the International Conference on Neural Networks, Volume 4, pp. 1942- 1948, 1995.

[8] IBM: An architectural blueprint for autonomic computing. Tech. rep., 2005. http://www-03.ibm.com/autonomic/pdfs/AC%20Blueprint%20White%20Paper%20V7.pdf