



FOODITY

Annex 1.1 Technical Annex

FOODITY - Open Call #2



This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement n° 101086105. This document reflects only the author's view, and the EU Commission is not responsible for any use that may be made of the information it contains.



Purpose of Annex 1.1 – Technical Annex

The purpose of this annex is to provide applicants with an understanding of the main technical components and infrastructure that they may be required to use as part of their applications. Please refer to **Annex 1 – Guidelines for Applicants** for specific details on what components and infrastructure are mandatory and in what cases.

This annex is structured into the following sections:

1. **DataU solution:** a cloud-based service for GDPR compliant data sharing between citizens and new applications and solutions from the open call sub-grantees of the FOODITY project.
2. **FOODITY datalake:** a centralised repository for storing and analysing large volumes of data related to the food and nutrition domains. It enables the collection and use of food-related data while safeguarding privacy rights and complying with GDPR regulations.
3. **FOODITY Component Library:** consists of the five components that the FOODITY project is making available to applicants in the framework of Open Call #2.



1. DataU solution

1.1. What is dataU?

The dataU is a GDPR compliant, cloud-based distributed platform for secure data-sharing which allows full control over data, who it is being shared with, and for how long. The service can be used in the context of sharing data between citizens (data owners) and third-party services (data processors), or alternatively between two companies, in accordance with their non-disclosure agreements.

DataU is viable in cases where there is personal or sensitive information involved, which requires permission to share and has a high level of trust and confidentiality. The platform serves as a communication layer between the parties using blockchain technology and does not store the data but grants and revokes access to it, and as such can be treated as a consent management platform. The storage of the data is therefore facilitated by the data processors, for as long as the data owners have permitted. The benefit for both parties is that the permissions are traceable and fully open to all involved so that conflicts and the associated risks are a thing of the past.

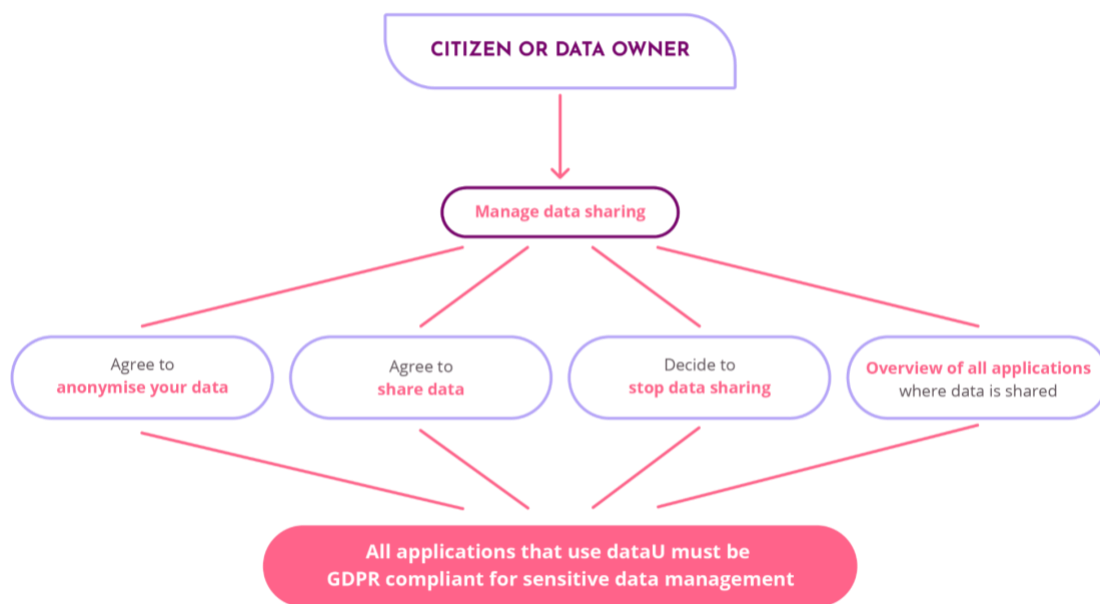


Figure 1. Different options available through use of the dataU.

1.2. How does dataU work?

The sharing of data is done in an encrypted, fully distributed, and decentralised manner, utilising the nodeU application, which exposes an API through which consent handling is realised. Data transport is done using an application that runs on the data processor's premises, called proxyU – responsible for setting up a peer-to-peer, secure tunnel to another proxyU after verifying the correct authentication, authorisation, and other parameters through a nodeU instance.

Both data processors and data owners are identified using their public keys, and verified using a certificate, which is included in every dataU application for the chain of trust to be validated. Data owners interact with dataU through a client application but there is, by design, no single mandatory application.



Instead, nodeU exposes an API through which web-based dashboards or native mobile or desktop applications can be developed. The FOODITY project will develop dashboardU as a reference implementation of such a client application.

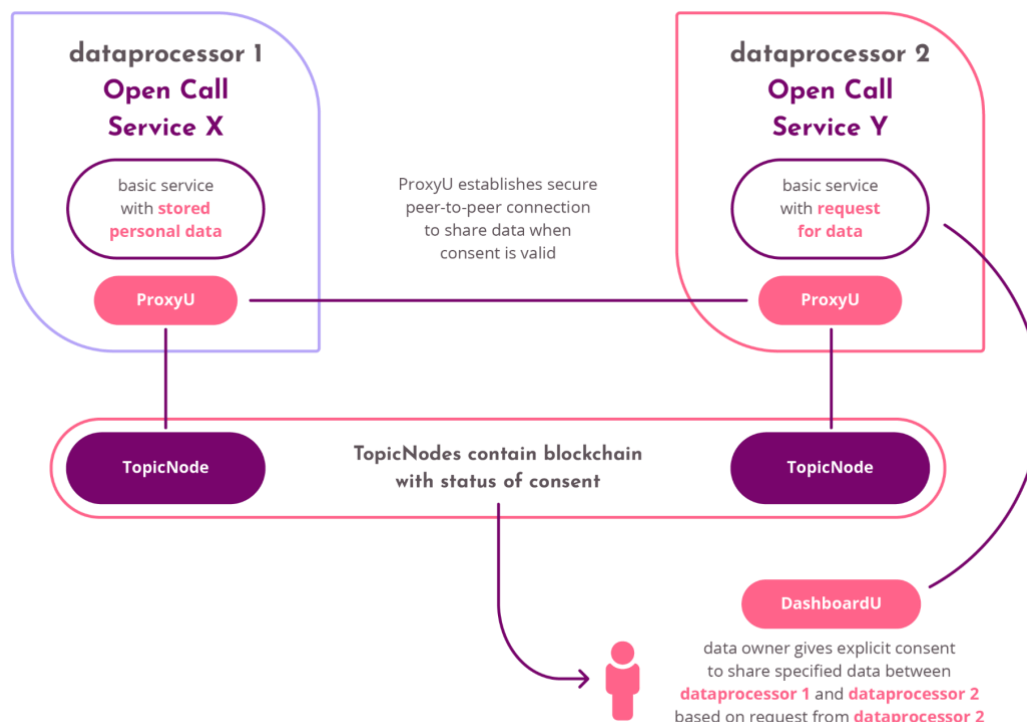


Figure 2. Visual representation of dataU application in terms of data flow between different processors.

2. FOODITY datalake

2.1. Purpose of the FOODITY datalake

The FOODITY datalake is a service for you and your data, providing a centralised repository that makes data accessible for stakeholders and researchers, whilst also giving you the opportunity to perform different analyses, from temporal trends to user behaviours. It encourages collaboration by connecting various entities, such as food enthusiasts and chefs, with businesses seeking innovations. The datalake offers efficient data search using Elastic search and user-friendly dashboards like Kibana. All of the data that enter adhere to GDPR compliance, ensuring citizen protection whilst enabling valuable insights for applications in areas including business intelligence and analytics.

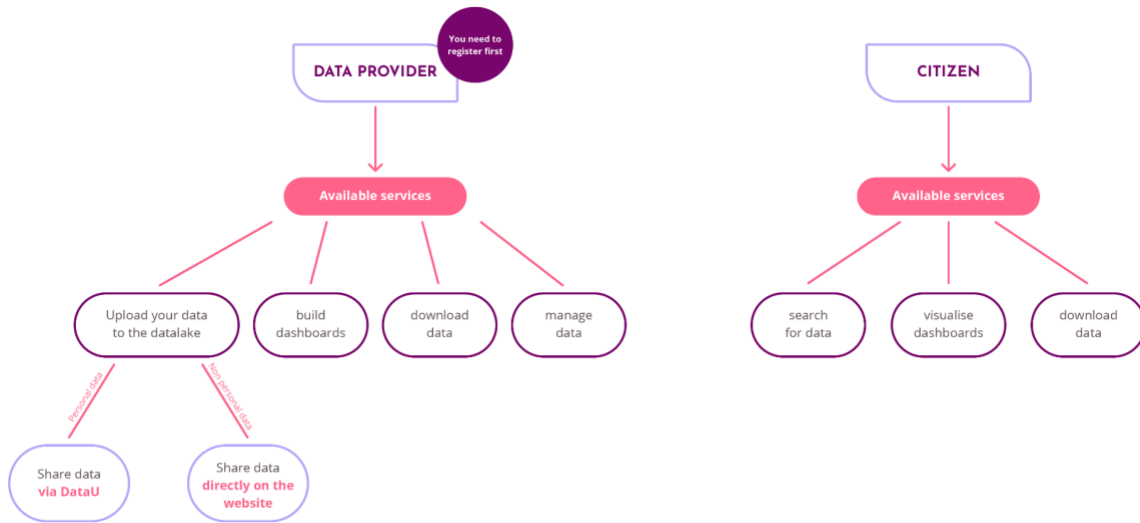


Figure 3. Visual representation of the datalake for data provider and citizen users.

2.2. How to upload the datasets into the datalake?

Data providers, including manufacturers and nutritionists, can submit data to the datalake. Before sharing datasets, providers must register on the website, complete their profiles, and provide metadata for their datasets. The access to the website is secured using advanced technology that ensures data privacy, prevents unauthorised access, and empowers data providers with control over their datasets.

For datasets containing personal or sensitive information, such as identifiable data, the DataU (our data consent management platform), comes into play to ensure that user consent is obtained before any data sharing occurs. On the other hand, non-personal data can be published directly on the FOODITY website, streamlining the process for datasets that do not require explicit user consent through DataU. To ensure GDPR compliance, data providers must follow guidelines such as data anonymisation, pseudonymisation, and minimisation.

2.3. Who can access the datalake?

European citizens can freely access the public dataset through the dedicated website. Authentication is not required for accessing public information about data providers and their datasets. Users can also manage their data through DataU if needed



3. FOODITY Component Library

3.1. Component #2: Food recognition from photo

Automatic recognition of food from a photo

What is provided? An AI model¹, i.e., a deep learning network that automatically recognises food from photos.

In what form is it provided? A .py (Python) file and .bin (model weights) file.

What does it do? Receives as input a photo of food, analyses it and provides as output a probability for each element in a specified list of foods (food titles), to be the depicted food.

How does it work? The model is based on a transformer network architecture which splits photos in patches, processes the patches in different scales, produces photo specific feature vectors and classifies them to food classes (specific foods). The model has been trained on and evaluated with two publicly available datasets for food recognition: Food-101 and Food2K (Figure 1). The Food-101 dataset consists of 101 food categories with 750 training and 250 test images per category, making a total of 101k images. Food2K is a large food recognition dataset with 2,000 categories and over 1 million images.

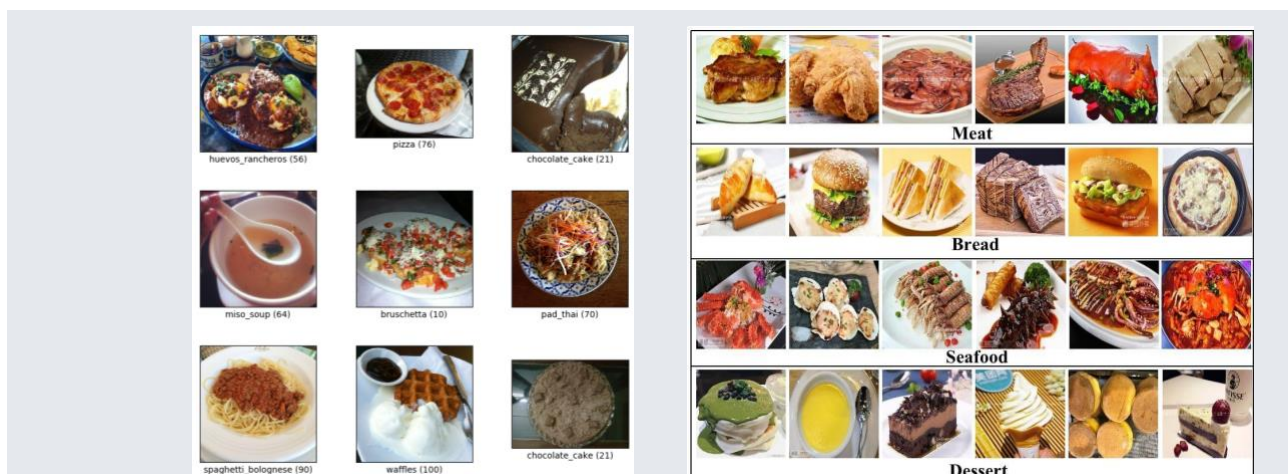


Figure 1. Examples of images contained in the Food-101 (left) and the Food2K (right) datasets.

Technical details/considerations. The above model has been developed, trained, and tested on the following system requirements (it is suggested that at least the above H/W requirements are met before using the component):

Hardware

- GPU = GeForce RTX 3080 10GB
- RAM = 32 GB
- CPU = i9-10900K

Software

- Cuda = 11.3
- Cudatoolkit = 11.3.1
- Python = 3.9.16
- Pytorch = 1.10.2

¹ An AI model is a tool or algorithm which is based on a certain data set through which it can arrive at a decision – all without the need for human interference in the decision-making process.



- Ubuntu 20.04.5 LTS

Additional to the code, a requirements.txt file for Conda (or similar) will be included so that an appropriate virtual environment can be created.

This model has not been tested on any mobile device. Therefore, for any use of this component on mobile devices, potential applicants are responsible for required adaptation.

Related datasets/standards. The datasets used for the training and test of this model are the Food-101² and Food2K³. They consist of images categorised in 101 categories and 2000 categories respectively. Some of the Food-101 categories are: apple_pie, baby_back_ribs, baklava, while some of the Food2K categories are: Margherita pizza, Black pepper steak, Tonkatsu.

IPR considerations. The component is provided under the following licence:

"Food recognition from photo" © 2023 by [Visual Computing Laboratory \(VCL\), Information Technologies Institute \(ITI\), Centre for Research and Technology Hellas \(CERTH\)](#) is licensed under [CC BY-NC 4.0](#)

The Food-101 dataset was introduced in [3]. Further IPR information can be found at. <https://www.kaggle.com/datasets/dansbecker/food-101>.

The Food2K dataset was introduced in [4].

² <https://www.kaggle.com/datasets/dansbecker/food-101>

³ <http://123.57.42.89/FoodProject.html>



3.2. Component #2: Food product identification from photo

Automatic food product identification from a photo of the product barcode.

What is provided? A python script that automatically identifies a food product barcode from a photo of the barcode and uses this information to retrieve product information from free online product databases.

In what form is it provided? A python script.

What does it do? Automatically identifies a food product barcode from a photo of the barcode, e.g., photo of a food product, and uses this information to retrieve food product information from free online databases.

How does it work? Given an image of a barcode (e.g., photo, screenshot, etc.) the component automatically identifies the barcode and connects to free online databases (e.g., the Open Food Facts⁴ database) via the corresponding APIs and retrieves the corresponding product information (Figure 2).

```

code: "8422410133482"
product:
  _id: "8422410133482"
  _keywords:
    0: "bonpreu"
    1: "maduixa"
    2: "refresc"
  added_countries_tags: []
  additives_old_tags: []
  additives_original_tags: []
  additives_tags: []
  allergens: ""
  allergens_from_ingredients: ""
  allergens_from_user: "(es) "
  allergens_hierarchy: []
  allergens_lc: "es"
  allergens_tags: []
  amino_acids_tags: []
  brands: "bonpreu"
  brands_tags:
    0: "bonpreu"

```

Figure 2. Screenshot of the json file retrieved after calling the Open Food Fact API. In the json file, information regarding a corresponding product's barcode is listed.

Technical details/considerations. The requirements of this component are:

- python 3.6
- pip 23.1.2

⁴ <https://world.openfoodfacts.org/>



Additionally, the component needs the following libraries to work:

- requests 2.31.0
- pyzbar 0.1.9
- PIL 10.0.0

Related datasets/standards. The component connects to the Open Food Facts database⁵ of food products, which is based on free contributions by users. The database consists of over two million products with their corresponding barcode. However, the component can be adjusted to connect to other similar databases, such as USDA FoodData Central, Nutritionix, Edamam and Open Menus are, each with their respective APIs.

IPR considerations. The component is provided under the following marking:

"Food product identification from photo" © 2023 by [Visual Computing Laboratory \(VCL\), Information Technologies Institute \(ITI\), Centre for Research and Technology Hellas \(CERTH\)](#) is marked with [CC0 1.0](#)

⁵ <https://world.openfoodfacts.org>



3.3. Component #3: Personalised nutrition plan

Automatic creation of a weekly personalised nutrition plan based on user profile information.

What is provided? A software component that recommends a weekly nutrition plan tailored to an individual's profile.

In what form is it provided? A Docker container.

What does it do? Receives as input a database of dishes, meals and user profiles and provides as output a weekly nutrition plan tailored to each user's profile.

How does it work? Initially, the software creates a list of candidate daily nutrition plans from the dishes and meals in the database (including kcal, micronutrients, ingredients, and recipes). Then, it considers a user's profile (e.g., height, weight, allergies, dietary preferences, etc.) and filters out daily nutrition plans based on the user profile restrictions (e.g., specific allergy). Subsequently, based on the user profile information, it calculates the appropriate daily caloric intake for the user and compares it to the caloric content of each daily nutrition plan. From this analysis, it ranks the daily nutrition plans from the most suitable to the least suitable. To ensure variety and avoid repetition, the algorithm also checks for meal diversity. Ultimately, the algorithm produces a personalised list of 7 daily nutrition plans (weekly nutrition plan) that are tailored to the user's specific profile and include diverse options. The component was based on validated research that has been reviewed and published [1].

Technical details/considerations. The recommendation system for generating meal plans is developed in Python 3.6 and integrated in a Django back-end and communicating with a React front-end. Therefore, a call with the corresponding API can trigger the process and return the meal plans.

Related datasets/standards. To use the component, two databases are required: a 'dishes' database and a 'meals' database. Below, the schema of the required DBs is described; Some basic content for filling (some of the required fields of) such databases can be found online at <https://zenodo.org/record/7308053> (PROTEIN NAP Database) or from other sources.

For the above recommendation system to work two databases are required.

1. The first database is the dishes database which consists of the below fields:

"ID": The unique identifier of the dish.

"Name (in local country language)": Name or short description of the dish.

"Name (in English)": Name or short description of the dish.

"Ingredients of a standard portion for a 3-6 year-old child (in local country language)": Text to be displayed for the ingredients of a standard portion of the dish for a child.

"Ingredients of a standard portion for a 3-6 year-old child (in English)": Text to be displayed for the ingredients of a standard portion of the dish for a child.

"Ingredients of a standard portion for a 7-12 year-old child (in local country language)": Text to be displayed for the ingredients of a standard portion of the dish for a child.



"Ingredients of a standard portion for a 7-12 year-old child (in English)": Text to be displayed for the ingredients of a standard portion of the dish for a child.

"Ingredients of a standard portion for a 13-15 year-old child (in local country language)": Text to be displayed for the ingredients of a standard portion of the dish for a child.

"Ingredients of a standard portion for a 13-15 year-old child (in English)": Text to be displayed for the ingredients of a standard portion of the dish for a child.

"Ingredients of a standard portion for a 16-18 year-old child (in local country language)": Text to be displayed for the ingredients of a standard portion of the dish for a child."

"Ingredients of a standard portion for a 16-18 year-old child (in English)": Text to be displayed for the ingredients of a standard portion of the dish for a child.

"Ingredients of a standard portion for an adult (in local country language)": Text to be displayed for the ingredients of a standard portion of the dish for an adult.

"Ingredients of a standard portion for an adult (in English)": Text to be displayed for the ingredients of a standard portion of the dish for an adult.

"Recipe (in local country language)": Text to be displayed for the dish recipe.

"Recipe (in English)": Text to be displayed for the dish recipe.

"Tip (in local country language)": Text to be displayed for tips associated with the recipe. OPTIONAL

"Tip (in English)": Text to be displayed for tips associated with the recipe. OPTIONAL

"Kcal": Kcal of a standard portion of the dish for an adult. Based on column 'M'

"Protein": Protein (in grams) of a standard portion of the dish for an adult. Based on column 'M'

"Fat": Fat (in grams) of a standard portion of the dish for an adult. Based on column 'M'

"Carbohydrates": Carbohydrates (in grams) of a standard portion of the dish for an adult. Based on column 'M'

"Dish type": The type of the dish.

"For Autumn": The dish is fine for the period Sept. 23 - Dec. 20.

"For Winter": The dish is fine for the period Dec. 21 - Mar. 20.

"For Spring": The dish is fine for the period Mar. 21 - Jun. 20.

"For Summer": The dish is fine for the period June. 21 - Sept. 22.

"White meat": The dish contains white meat.

"Red meat": The dish contains red meat, sausages or cured meat.

"Pork": The dish contains pork.

"Fish or seafood": The dish contains fish or seafood.

"Pulses (Legumes)": The dish contains pulses (legumes).

"Dairy": The dish contains dairy ingredients.



"Eggs": The dish contains eggs.

"Pasta": The dish contains pasta.

"Rice": The dish contains rice.

"Tubers": The dish contains tubers.

"Soups": The dish contains soups.

"Cereals": The dish contains cereals.

"Fruit": The dish contains fruit.

"Nuts": The dish contains nuts.

"Raw vegetables": The dish contains raw vegetables.

"Cooked vegetables": The dish contains cooked vegetables.

"Protein mix": The dish contains a protein mix.

"Unique": The dish is a unique dish.

"Semi Unique": The dish is a semi-unique dish.

"Vegetables for semi": The dish contains vegetables for semi-unique dishes.

"Red": The dish contains red vegetables.

"Green": The dish contains green vegetables.

"White": The dish contains white vegetables.

"Yellow": The dish contains yellow/orange vegetables.

"Purple": The dish contains purple vegetables.

"Multicolor": The dish contains multi-colour vegetables.

"No colour": The dish does not contain coloured vegetables.

"For proposals": The dish is fine for dinner proposals and for non-school day lunch proposals.

"For school": The dish is served at schools.

2. The second database is the meals database which consists by the following fields:

"ID The unique identifier of the dish.

"Name (in local country language)": Name or short description of the meal.

"Name (in English)": Name or short description of the meal.

"Type": The type of the meal.

"Country": The country at which the meal is aimed.

"For Autumn": The dish is fine for the period Sept. 1/Nov. 14.

"For Winter": The dish is fine for the period Nov. 15/Feb. 28.



"For Spring": The dish is fine for the period Mar. 1/May 14.

"For Summer": The dish is fine for the period May 15/Aug. 31.

"Dish ID #2": Compose your meal by selecting 1 up to 10 food IDs from the respective 'Dishes' tab.

"Dish ID #2"

"Dish ID #3"

"Dish ID #4"

"Dish ID #5"

"Dish ID #6"

"Dish ID #7"

"Dish ID #8"

"Dish ID #9"

"Dish ID #10"

It is made clear that the construction and filling of the database is needed by anyone who wishes to use the present component.

IPR considerations. The component is provided under the following licence:

"Personalised nutrition plan" © 2023 by [Visual Computing Laboratory \(VCL\), Information Technologies Institute \(ITI\), Centre for Research and Technology Hellas \(CERTH\)](#) is licensed under [CC BY-NC-SA 4.0](#)



3.4. Component #4: Personalised support for nutrition/physical activity

Personalised decision support for healthy diet and physical activity.

What is provided? A knowledge-based, decision support system (DSS) in the form of Software as a Service (SaaS), that prioritises food and physical activities according to the personal medical conditions, needs and preferences of an individual.

In what form is it provided? A Docker container.

What does it do? The system prioritises nutrition and physical activities (e.g., in a meal plan, restaurant menu, activity plan, etc.), based on whether they are advised given a user's personal medical conditions, needs and preferences. In addition, it filters out - or can potentially issue an alert - over given foods and physical activities when they are not medically advised or even potentially hazardous for a given user.

How does it work? Given the Nutrition & Activity Ontology (NAct) ontology, the knowledge based DSS matches (1) the semantic profile expressing the nutritional, medical and well-being needs, restrictions, and preferences of a user and (2) the semantic transcoding of a food's ingredients and/or an activity's properties, to:

1. Reject foods and/or physical activities that are incompatible with a patient's profile (e.g., would trigger one or more of their allergies or are restricted by doctors due to a medical condition).
2. Prioritise foods and/or physical activities based on the user's personal needs, conditions and/or preferences. For instance, a deficiency may require an increase in the intake of a certain nutrient, or a medical condition may mandate uptake of a particular type of exercise. All factors in the user's personal profile contribute simultaneously to the final prioritisation score of each food/activity.

The component was based on validated research that has been reviewed and published [1].

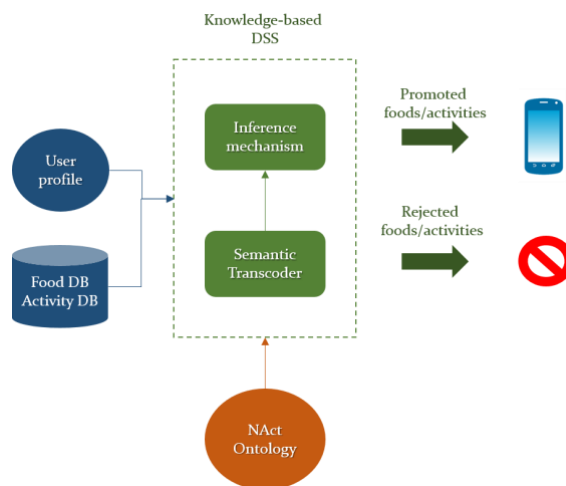


Figure 3. Mechanism of the Knowledge-based DSS

Technical details/considerations. The above reasoner is based on Lightweight Fuzzy semantic Reasoner (LiFR). LiFR is a fuzzy extension of the Pocket KRHyper reasoner, initially designed for first-generation mobile devices. Built on the hyper-tableaux calculus, LiFR performs first-order model generation and translates Description Logic (DL) axioms into first-order clauses. It extends Pocket



KRHyper to handle fuzziness, incorporating semantics and transformations for fuzzy operators. LiFR has improved efficiency and disjunction handling, making it suitable for limited-resource devices. It offers inferencing services such as consistency checking, satisfiability checking, concept subsumption, fuzzy entailment, and calculation of the Best Entailment Degree (BED). By generating models satisfying the fuzzy knowledge base, LiFR natively supports computing the BED for various combinations of individuals and concepts.

Related datasets/standards. For related datasets and standards see Component #5 below.

IPR considerations. The component is provided under the following licence:

“Personalised support for nutrition/physical activity” © 2023 by [Dorothea Tsatsou, Visual Computing Laboratory \(VCL\), Information Technologies Institute \(ITI\), Centre for Research and Technology Hellas \(CERTH\)](#) is licensed under [GNU Lesser General Public License v3.0](#)



3.5. Component #5: Nutrition and Activity Ontology

The Nutrition and Activity Ontology (NAct)

What is provided? An ontology, i.e., a formal expert model of nutritional, medical and lifestyle factors that affect healthy living.

In what form is it provided? A Web Ontology Language (OWL) project.

What does it do? It comprises an amalgamation of European guidelines relevant to nutrition and well-being. It also includes experts' evidence-based knowledge, pertaining to foods and their relation to nutrients as well as to foods and/or physical activities and their well-being consequences, restrictions and specific goals relevant to user conditions. It can be used to coalesce nutritional, medical, behavioural and lifestyle indicators with potential dietary and physical activity directives.

How does it work? Nutrition & Activity Ontology (NAct) ontology is used by the knowledge-based decision support system (DSS) as: (1) the finite vocabulary to describe the nutritional, medical and lifestyle characteristics of users (user profiling), as well as the nutritional and lifestyle attributes of foods and activities and (2) the axiomatic backbone by which the knowledge-based DSS infers personalised decisions for any user profile, given the attributes of available foods/activities. Since it comprises a standard ontology, modelling vast information over the healthy living domain, it may also be employed by any means of semantic technology (e.g., inference engine, query engine, semantic classifier, etc.) to retrieve information over such factors and/or drive inference for different contexts in the healthy living domain. The component was based on validated research that has been reviewed and published [2].

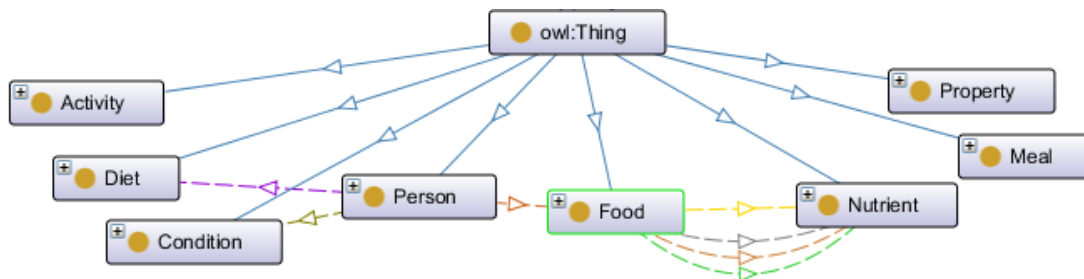


Figure 4. The top-level hierarchy of the NAct ontology

Technical details/considerations

NAct is an OWL ontology, falling in the OWL 2 RL expressivity fragment. Additionally, it has been engineered using the protégé⁶ ontology editor. NAct is available under a persistent PURL URI⁷. It is published on GitHub, in a dedicated project and repository⁸, while the ontology specification and documentation (LODE [8] version) web page⁹ will be permanently maintained through GitHub pages.

⁶ <https://protege.stanford.edu>

⁷ <http://purl.org/nact>

⁸ <https://github.com/nutritionactivityontology/nact>

⁹ <https://nutritionactivityontology.github.io/nact/>



Related datasets/standards

The above ontology is the NAct., which has been constructed based on European nutritional standards. More specifically the ontology is based on the works of McCance and Widdowson food database [5] and the Compendium of Physical Activities [6], which were deemed by the domain experts as the broadest and most adequate databases adhering to European nutritional, health and well-being standards.

Additionally, the standards of component's 5 ontology are based on the Methontology [7] methodology pertaining to the seven stages: specification, knowledge acquisition, conceptualisation, integration, implementation, evaluation, and documentation.

IPR considerations.

The component is provided under the following licence:

[The Nutrition & Activity Ontology \(NAct\)](#) © 2021 by [Dorothea Tsatsou, Visual Computing Laboratory \(VCL\), Information Technologies Institute \(ITI\), Centre for Research and Technology Hellas \(CERTH\)](#) is licensed under [CC BY-SA 4.0](#)



3.6. References

1. Stefanidis, K.; Tsatsou, D.; Konstantinidis, D.; Gymnopoulos, L.; Daras, P.; Wilson-Barnes, S.; Hart, K.; Cornelissen, V.; Decorte, E.; Lalama, E.; Pfeiffer, A.; Hassapidou, M.; Pagkalos, I.; Argiriou, A.; Rouskas, K.; Hadjidimitriou, S.; Charisis, V.; Dias, S.B.; Diniz, J.A.; Telo, G.; Silva, H.; Bensenousi, A.; Dimitropoulos, K. PROTEIN AI Advisor: A Knowledge-Based Recommendation Framework Using Expert-Validated Meals for Healthy Diets. *Nutrients* **2022**, *14*, 4435. <https://doi.org/10.3390/nu14204435>
2. Neuhaus, F., & Brodaric, B. (2022, January). Nact: The nutrition & activity ontology for healthy living. In *Formal Ontology in Information Systems: Proceedings of the Twelfth International Conference (FOIS 2021)* (Vol. 344, p. 129). IOS Press.
3. Bossard, L., Guillaumin, M., & Van Gool, L. (2014). Food-101–mining discriminative components with random forests. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part VI 13* (pp. 446–461). Springer International Publishing.
4. Min, W., Wang, Z., Liu, Y., Luo, M., Kang, L., Wei, X., ... & Jiang, S. (2023). Large scale visual food recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
5. McCance, R. A., & Widdowson, E. M. (2014). McCance and Widdowson's the Composition of Foods. Royal Society of Chemistry.
6. Ainsworth, B. E., Haskell, W. L., Herrmann, S. D., Meckes, N., Bassett Jr, D. R., Tudor-Locke, C., ... & Leon, A. S. (2011). 2011 Compendium of Physical Activities: a second update of codes and MET values. *Medicine & science in sports & exercise*, *43*(8), 1575–1581.
7. Fernández-López, M., Gómez-Pérez, A., & Juristo, N. (1997). Methontology: from ontological art towards ontological engineering.
8. Peroni, S., Shotton, D. M., & Vitali, F. (2012). Making Ontology Documentation with LOD. In *I-SEMANTICS (Posters & Demos)* (pp. 63–67)