D.7.3 Integrated Service Environment

E. Liu, Y. Zhao, G. Drosatos, N. Portokallidis, E. Kaldoudi, H. Wei, A. Third,
V. Marozas, A. Sološenko, S. Daukantas, T. Smagurauskas, J. Piwinski, G.
Clapworthy

June 2016

CARRE Contacts

Project Coordinator:	Eleni Kaldoudi	kaldoudi@med.duth.gr
Project Manager:	George Drosatos	gdrosato@ee.duth.gr

DUTH Democritus University of Thrace	Eleni Kaldoudi	kaldoudi@med.duth.gr
OU The Open University	John Domingue	john.domingue@open.ac.uk
BED Bedfordshire University	Enjie Liu	enjie.liu@beds.ac.uk
VULSK Vilnius University Hospital Santariškių Klinikos	Domantas Stundys	domantas.stundys@santa.lt
KTU Kaunas University of Technology	Arunas Lukosevicius	arunas.lukosevicius@ktu.lt
PIAP Industrial Research Institute for Automation & Measurements	Roman Szewczyk	rszewczyk@piap.pl

Disclaimer

Document Control Page

Project

Contract No.: 611140

Acronym: CARRE

Title: Personalized Patient Empowerment and Shared Decision Support for Cardiorenal Disease and Comorbidities

Type: STREP

Start: 1 November 2013

End: 31 October 2016

Programme: FP7-ICT-2013.5.1

Website: http://www.carre-project.eu/

Deliverable

Deliverable No.: D.7.3

Deliverable Title: Integrated Service Environment

Responsible Partner: BED (Dr. Enjie Liu)

Authors: E. Liu, Y. Zhao, G. Drosatos, N. Portokallidis, E. Kaldoudi, H. Wei, A. Third, V. Marozas, A. Solosenko, S. Daukantas, T. Smagurauskas, J. Piwinski, G. Clapworthy

Input from: All partners

Peer Reviewers: D. Jegelevicius (KTU), A. Third (OU)

Task: T.7.3 System and Service Integration

Task Duration: 9 months: 1 September 2015 to 30 June 2016

Work Package: WP7: Integration and evaluation

Work Package Leader: VULSK – Romualdas Kizlaitis

Due Date: 30 June 2016

Actual Delivery Date: 14 July 2016

Dissemination Level: PU

Nature: R & D

Files and format: Deliverable report: 1 pdf file
Software: available from project site https://www.carre-project.eu/innovation/carre-system-integration/

Version: 16

Status:
☐ Draft
☒ Consortium reviewed
☒ WP leader accepted
☒ Coordinator accepted
☐ EC accepted

Document Revision History

Version	Date	Modifications	Contributors
v01.0	01 June 2016	Initial version	Enjie Liu, Youbing Zhao
v02.0	09 June 2016	Version after the skype meeting	George Drosatos, Nick Portokallidis, Allan Third, Hui Wei
v03.0	15 June 2016	Updated version after skype meeting	Eleni Kaldoudi, George Drosatos, Nick Portokallidis
V04.9	18 June 2016	Further updated version	Enjie Liu, Youbing
v07.1	24 June 2016	With partners' contribution	Vaidotas Marozas, Tomas Smagurauskas
v08.0	29 June 2016	With partner's contribution	Allan Third
v09.0	29 June 2016	Updated version	Youbing Zhao
v11.0	30 June 2016	Updated version	Jan Piwinski, Youbing Zhao, Enjie Liu
v13.0	1 July 2016	Updated version	Youbing Zhao, Vadotas Marozas
v14.0	2 July 2016	Updated version	Youbing Zhao, Enjie Liu
v14.2	3 July 2016	Updated version	Youbing Zhao, Enjie Liu
v14.3	4 July 2016	Update based on KTU's review	Vaidotas Marozas, Youbing Zhao, Enjie Liu
v14.5	4 July 2016	Update based on OU's review	Allan Third, Youbing Zhao, Enjie Liu
v14.7	5 July 2016	Update data integration and access control sections	George Drosatos, Nick Portokallidis
v14.9	5 July 2016	Update based on DUTH's review	George Drosatos, Nick Portokallidis, , Youbing Zhao, Enjie Liu
v15.0	6 July 2016	Proofreading	Youbing Zhao, Enjie Liu
v15.2	13 July 2016	Final checking	Gordon Clapworthy
V16.0	14 July 2016	Final version	G. Drosatos, E.Kaldoudi

Table of Contents

List of Figures

List of Tables

Executive Summary

In CARRE, the personalised patient-empowerment and decision-support services require the complex collection, presentation and analysis of a large volume of heterogeneous data from a variety of data sources including sensors, risk factors, PHR, decision support, etc., and involve various sets of components from a range of different modules. Properly planned integration makes it possible to organise and coordinate these components to work in unison to provide the planned system tasks.

The goal of Workpackage 7 is the integration of all modules developed in the previous work packages into a personalized services environment for patients as well as medical professionals and its evaluation, including the establishment of appropriate processing chains, workflows and the required middleware to allow intra-component communication.

This document is a report of deliverable D7.3 "Integrated Service Environment" and describes how the components of the system are organised to communicate and interact with each other to provide planned system functions. This component- and service-based integration also has the potential for exposing the CARRE service at the system- or component-level for external exploitation and extension in the future.

About CARRE

CARRE is a project, funded by the FP7 Programme of the European Commission, which will provide an innovative means for managing comorbidities (multiple co-occurring medical conditions), with an immediate focus on chronic cardiac and renal disease patients or persons with an increased risk of such conditions.

Sources of medical and other knowledge will be semantically linked with outputs from sensors to provide clinical information, personalised to the individual patient that can help to support the tracking of the progression and interactions of comorbid conditions. Visual analytics will be employed so that patients and clinicians will be able to visualise, understand and interact with this linked knowledge and take advantage of personalised empowerment services supported by a dedicated decision-support system.

The ultimate goal is to provide, via an integrative approach, a means by which patients with comorbidities are helped to take an active role in care processes, including self-care and shared decision-making, and medical professionals are supported in understanding and treating comorbidities.

Terms and Definitions

The following are definitions of terms, abbreviations and acronyms used in this document.

Term	Definition
AF	Atrial fibrillation
API	Application programming interface (API) is a set of functions and procedures that allow the creation of applications that access the features or data of an operating system, application, web services or other service
DSS	Decision Support Service
EC	European Commission
ECG	Electrocardiogram
EHC	European Health Card
eHealth	Electronic Health
EU	European Union
HTML	Hypertext Markup Language, the standard markup language used to create web pages.
HTTP	Hypertext Transfer Protocol, a data communication for the World Wide Web.
HTTPS	Hypertext Transfer Protocol Secure
OAuth	OAuth is an open standard for authorization, commonly used as a way for Internet users to log in to third party websites
OWL	Web Ontology Language
PHR	Personal Health Record
REST	Representational State Transfer is a software architecture style for building scalable web services.
RDF	Resource Description Framework - a standard model for data interchange on the Web.
SPARQL	RDF query language, that is, a query language for databases, able to retrieve and manipulate data stored in RDF
SVG	Scalar Vector Graphics, is an XML-based vector image format for two-dimensional graphics with support for interactivity and animation
UC	Use Case
UML	Unified Modeling Language
UMLS	Unified Medical Language System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VA	Visual Analytics
Vis	Visualisation
WWW	World Wide Web

1. Introduction

CARRE targets personalized patient empowerment and decision support based on risk models from the medical literature and measurement data from patients. CARRE is a system that involves multiple data sources, data processing and analysis modules and tries to provide a variety of visual presentation, analysis and decision making functions. It comprises a set of components or modules which are functionally distinguishable but which interact with each other in a coordinated way during execution to provide the system functions. Without proper integration, these components will not work in unison to achieve the goals. Figure 1 shows the concept diagram of the CARRE system from the DoW.
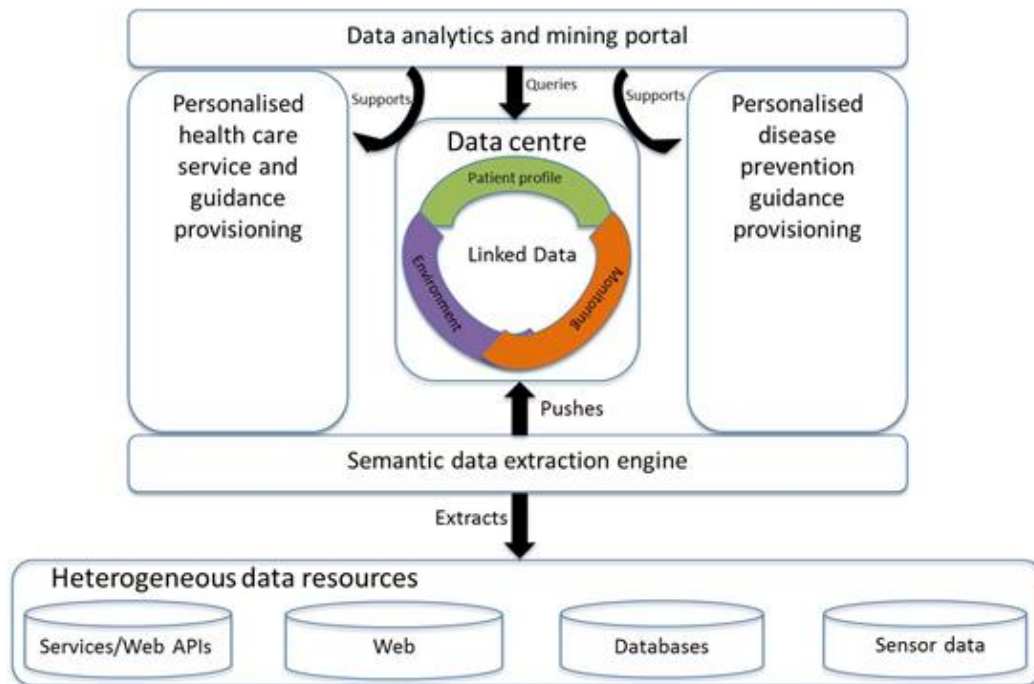
Figure 1. CARRE concept diagram (from DoW)

For data-oriented functions, the CARRE system has to provide:

1. Data aggregator components for collecting personalised health data from heterogeneous data sources, such as sensors, PHR, medical literatures, etc.
2. A semantic data extraction engine which adopts a crawling approach to harvest data from heterogeneous resources and extracts them using Linked Data principles into the Linked-Data-based repository.
3. A Linked-Data-based data centre which can efficiently integrate all the necessary types of data, adding internal semantic links among them as well as external semantic links to Linked Open Data knowledge. In additional, the data centre will support the query endpoints for semantically retrieving the data.

For user-oriented functions, the CARRE system needs to provide:

1. Personalised service and treatment guidance that will deliver treatment guidance, alerts and education suitable to the individual patient's needs.
2. Disease-prevention guidance that will provide personalised advice on life-style choices to help with the management of risks for comorbidities or the progression of the diseases.
3. Visual analytics and data mining user interfaces to support access to visual analytics and data mining tools for use both by patients and by medical professionals.
4. User interfaces for medical professionals to manually enter the risk factors and user interfaces for patients to enter their PHR data.

In summary, the CARRE system collects, extracts and stores data from a variety of heterogeneous data sources including sensors, PHR, risk factors, decision support, etc.; this requires the integration of various data access control, collection, processing and storage components. It also provides a range of user-oriented interfaces,

such as visual analytics, risk entry, PHR entry, risk alert visualisation, etc., and these must be integrated with the access control and the data repository components to provide the overall CARRE infrastructure.

This deliverable describes the integration, performed in Task T7.3, *System and Service Integration*, of modules developed in previous work packages by different partners and their deployment for evaluation. The integration is realised in a loosely coupled, interoperable and extensible way. The data repository is exposed using standards-based RESTful web services, by use of which any other components can access and store data. Benefitting from the shared-access-control infrastructure, the main visual interface only provides links to the user interfaces of the other components. This component- and service-based integration achieves flexible and reliable integration of the CARRE system; however, it also provides the potential for exposing the CARRE service at the system- or component-level, when possible external exploitation and further extensions are considered in the future.

The remainder of this deliverable report is organised as follows. Section 2 reviews the overall architecture of the CARRE system and the approach taken to the integration. With a focus on the infrastructural services, Section 3 introduces the integration of the public and private data repositories, and Section 4 presents the integration of access control and user authentication. In relation to the visual presentation layers, Section 5 describes details of the integration of different CARRE components for the visual interface at component level, infrastructural level and the application level, while the techniques used for a systematic localisation are covered in Section 6. Section 7 summarises the technical requirements necessary for the deployment of the integrated system, and Section 8 draws some conclusions.

2. Overview of CARRE system integration

2.1. CARRE system architecture and components

The CARRE high level architecture design in Figure 2 was proposed in D2.5[1] and has been implemented and reported on by the corresponding partners at the component level during the course of project development. However the connections, interactions, communications and dependencies between these components, which are critical in orchestrating them to present the live CARRE system functions and services, remain almost uncovered because the system-level connections and interactions can hardly be covered comprehensively and systematically by a single component. This deliverable on system integration reports on the missing information of inter-component coordinations, interactions, communications and dependencies at the system level and presents the full spectrum of the mechanisms of the CARRE system.

As shown in Figure 2, the CARRE components are designed to facilitate the realisation of the CARRE concept. The public and private data aggregators are used to extract heterogeneous data semantically from various resources, and the public and private RDF[2] repositories form the data centre, which is based on Linked Data; CARRE services provide personalised services and treatment guidance as well as personalised disease-prevention guidance to individual patients; CARRE end-user applications provide functions that will enhance and extend the services into the future; the CARRE visual interface provides an integrated user interface to access the CARRE services and end-user applications.

---

[1]   D2.5 CARRE architecture, https://www.carre-project.eu/project-info/deliverables/, accessed on 28/06/2016

[2]   https://www.w3.org/RDF/
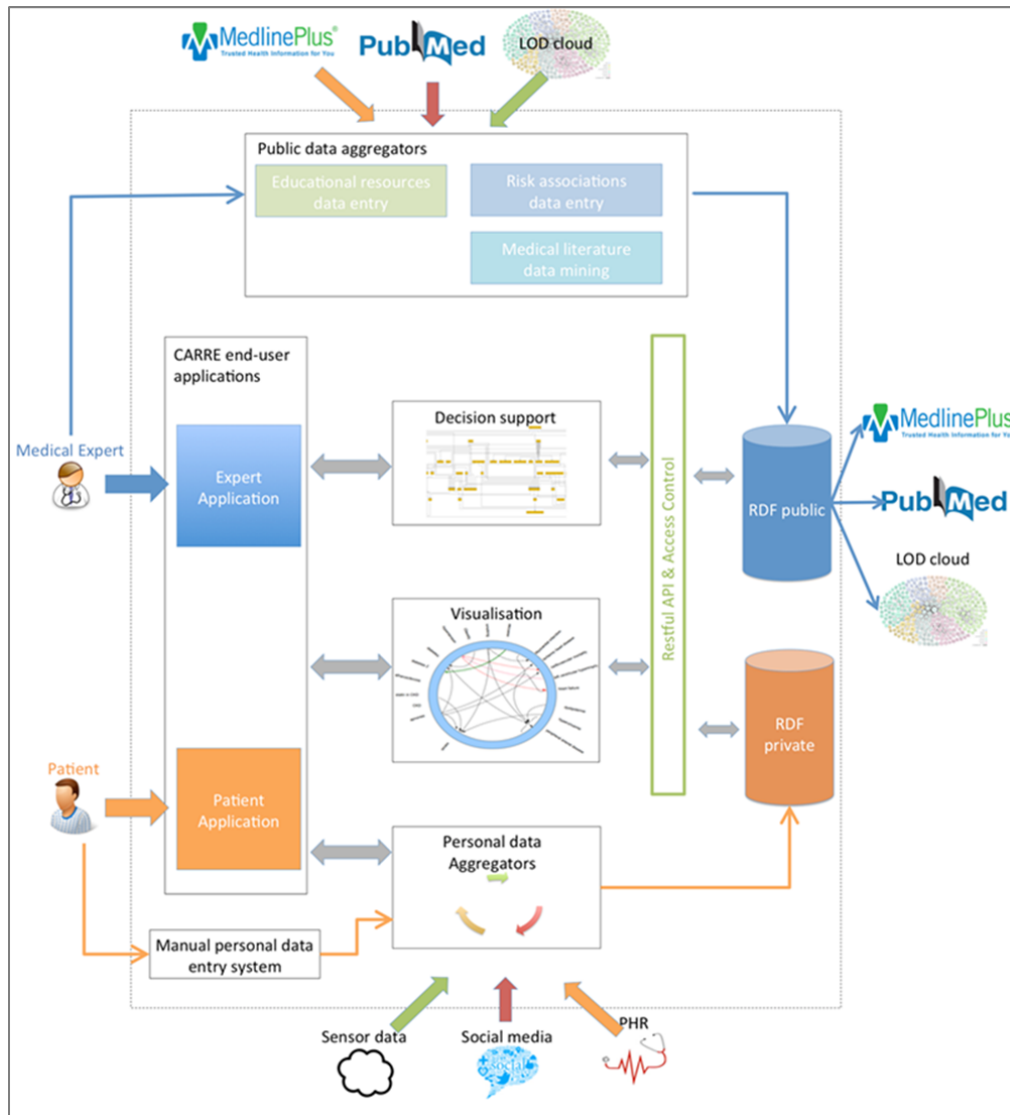
Figure 2. High-level CARRE system architecture

2.1.1. Public data aggregation

Data mining of the medical literature: Medical evidence aggregators mine medical knowledge from the electronic medical literature with the aim of 1) enriching the evidence of the existing risk descriptions as entered manually by medical experts, and 2) identifying new risk associations for cardiorenal diseases and comorbidity published during and beyond the project's lifetime. Once evaluated and validated by medical professionals, new risk associations can be manually input into the CARRE public RDF repository via the risk entry system.

Aggregator of educational resources: This module aggregates the metadata of educational resources available in third-party repositories on the web. Data aggregation consumes the existing APIs of educational resource repositories whose metadata is then stored as RDF triples in the CARRE public repository.

Risk factor manual entry system: This risk entry system enables the medical experts to encode the risk factors between biological, demographic, lifestyle and environmental elements and clinical outcomes in accordance with evidence from the clinical literature.

2.1.2. Private data aggregation

Sensor data aggregators: This component retrieves sensor data from sensor dataclouds and deposits it into the private RDF repository. Authorisation to connect to the sensor datacloud is gained from from the end-users and the sensor data is then imported by fetching historical and onging data, where available. The aggregators are also responsible for transforming this data into the appropriate RDF triples that conform to the CARRE ontology.

PHR aggregators: PHR aggregators are responsible for acquiring permission from the users to access third-party PHR systems (which provide APIs for accessing their records). The aggregators then fetch the relevant information from personal records, convert the data into RDF and deposit them into the private RDF repository.

PHR manual entry system: This browser-based application allows patients to record their medical data by entering them manually through web forms. It is designed to complement the PHR aggregator when either a PHR is not available or extra data needs to be input.

Web lifestyle data aggregator: This collects information on the user's intentions, mainly on the following topics: the patient's inclination to search for health-related information on the web and the types of the health-related information (s)he is interested in. It captures lifestyle data, extracts the user's intentions and stores it into the private RDF repository.

ECG aggregator: This is integrated into the *Web lifestyle data aggregator* and is intended for heart arrhythmia (atrial fibrillation - AF) detection by analysing electrocardiogram (ECG) signals recorded with Emotion Faros (Mega Electronics Ltd, Finland) devices. The aggregator performs AF parametrization from the start time and duration of the AF episode and the average heart rate, and stores these parameters into the private RDF repository.

2.1.3. RDF repositories

Private RDF repository: This repository implements a set of secure mechanisms for ensuring the user's privacy and data security and conforms to the CARRE ontologies. It uses two separate graphs:

- *Patient private graph*: this stores the personal data (sensor data, PHR and web lifestyle data) and generated data (i.e. personalized DSS alerts).
- *Medical expert private graph*: this stores data relevant to the patient's management.

Public RDF repository: The data hosted here originate from medical experts and the medical evidence aggregators, converted to conform to the CARRE ontologies. It has three graphs that cover specific needs:

- *Risk data graph*: stores risk factors in accordance with evidence from the clinical literature.
- *Educational resources graph*: stores educational resources relevant to diseaces.
- *DSS data graph*: stores DSS algorithms and alerts for patients and medical experts.

2.1.4. CARRE services

Visualisation component: This includes interactive functions to visualize and analyze the data stored in the public and private data repository, providing visual presentations to help users to understand the personal health status, the disease progressions etc.

Decision support service: This provides ontology-based reasoning – it produces meaningful information that is passed to the end-user interface with the synergy of the visualisation component.

Access control component: It implements CARRE users' authentication (patients and medical experts) and provides access control to RDF repositories by authorizing read/write permissions. The access control is achieved and enforced via the RESTful API.

2.1.5. CARRE end-user applications

DSS alerts entry system: This allows the medical experts and patients to adapt existing decision support services or build new ones.

Management system for the personal aggregators: This system implements the connection of sensors' manufacturers and PHR providers with a private RDF repository. This platform allows patients to control what personal data are uploaded to their private RDF and from which manufacturer/provider.

Visual interface component: This provides a graphical interface for users to access the end-user applications and services provided by other components. The interfaces for patients and for medical professionals are different.

2.2. Integration approaches

In CARRE, most of components are web-based and many functions, such as the RDF data repositories, medical and educational data mining service, are exposed as APIs or Web Services. Thus, in general, the integration of CARRE components is web-based and can take place at one of three layers of an application[3]:

- User Interface
- Application Functionality
- Data Access

Integration at the Application Functionality layer is a traditional approach for programmatic access to an application, and can be achieved by integrating with an API or using web services. It provides direct access to the business logic of an application. However, when an API is not available, the integration can be achieved only by modifying the application code itself, leading to the undesirable situation that when the underlying code changes, the integration may break. Hence, integration at the application level is not a flexible and durable solution, especially when the application code may be revised in the future. In CARRE, the web integration takes place in two layers: the user interface layer and the data access layer to promote a uniform programmatic interface for all applications.

2.2.1. Data integration

The CARRE system has a central RDF repository, which includes public and private data as already described in Section 2.1 and fuurther detailed in Section 3. Table 1 summarises the data that are aggregated in CARRE and the approaches that are used to collect them.

Integration at the Data Access layer is the most efficient approach to obtaining data from various applications and it is typically achieved through connections directly to the underlying data resources, which by-passes the business logic of the application. It is only suitable for data inquiry and aggregation requirements. However, by utilizing the applications transaction logic, for functions such as updates, the integrity of the new data being added or modified can be ensured.

It is worth noting that data collection is a continuing task. In the case of collecting data from CARRE services or the Web, as showed in Table 1, each type of data comes with a search mechanism, such as medical and educational evidence aggregation interface; and each type of data also comes with an automatic or manual system, such as the risk factor manual entry system. For example, the manual medical evidence entry system is in place for adding new risk associations after the evidence is validated by medical professionals.

In the case of collecting data from the third-party database, in particular collecting data from personal monitoring devices, such as sensors that are used in CARRE, e.g., Fitbit, Withings, etc., the details of the secured authentication approach are given in section 4.

---

[3] Web Integration Technologies, RedOak software, http://www.redoaksw.com/products/webclipper/RedOak_WebIntegrationOverview.pdf, accessed on 28/06/2016

Table 1. Summary of CARRE data and their integration approaches.		
Source of CARRE data	**Collection approach**	**CARRE Repository**
CARRE Service		
Decision Support (DSS)	Via DSS alert system, the data is pre-stored or manually input via DSS manual alert input system	public & private
Web		
Medical literature data mining	Data search is performed via the medical evidence aggregation interface, and the confirmed evidence is input by doctors via the risk factor manual entry system	public
Educational resources aggregator	Data search is performed via the educational resources aggregator interface and the collected metadata can be added to the public repository	public
3nd-party Database		
Personal sensor data	Retrieved from relevant vendor datacloud	private
PHR	Retrieved from third-party PHR systems	private

2.2.2. User interface integration

Web Integration turns the browser into a well-defined programmatic interface that exposes the full functionality and data of the application. The Web Integration Server accesses the presentation layer of an application through the browser interface and can then make one or more of the following access points available: a web service, a modified browser interface, or an API into the application's logic and data.

Integration from the user interface layer makes distributed deployment easier, since the integration developers do not need to understand the functional modules within each application. The key function that the web integration needs to provide is a standard and universal API that processes the business logic. The physical web application server is transparent to the end-user who operates on the main web interface. Knowing the necessary parameters and access of the user's credentials, the business logic and data repository, allows the web interface to identify the next page to visit.

Figure 3 illustrates the general concept of the CARRE user interface integration. The CARRE applications referred in the figure include the DSS alert manual input system, the medical evidence aggregator, the risk factor manual entry system, the educational resources aggregator, the educational metadata entry system, and the CARRE devices system.

Figure 4 uses sequence diagram to show the interface integration.

Sections 3 and 5 below explain the two types of integration in CARRE, respectively. In Section 4, we briefly summarise the user access control mechanisms in CARRE, the detailed implementation has been presented in the previous deliverables. Section 6 dedicates to the implementation of localisation in CARRE, which plays a crucial role in deploying CARRE system to a larger user community at a European level. Section 7 provides detailed information on the CARRE system deployment.
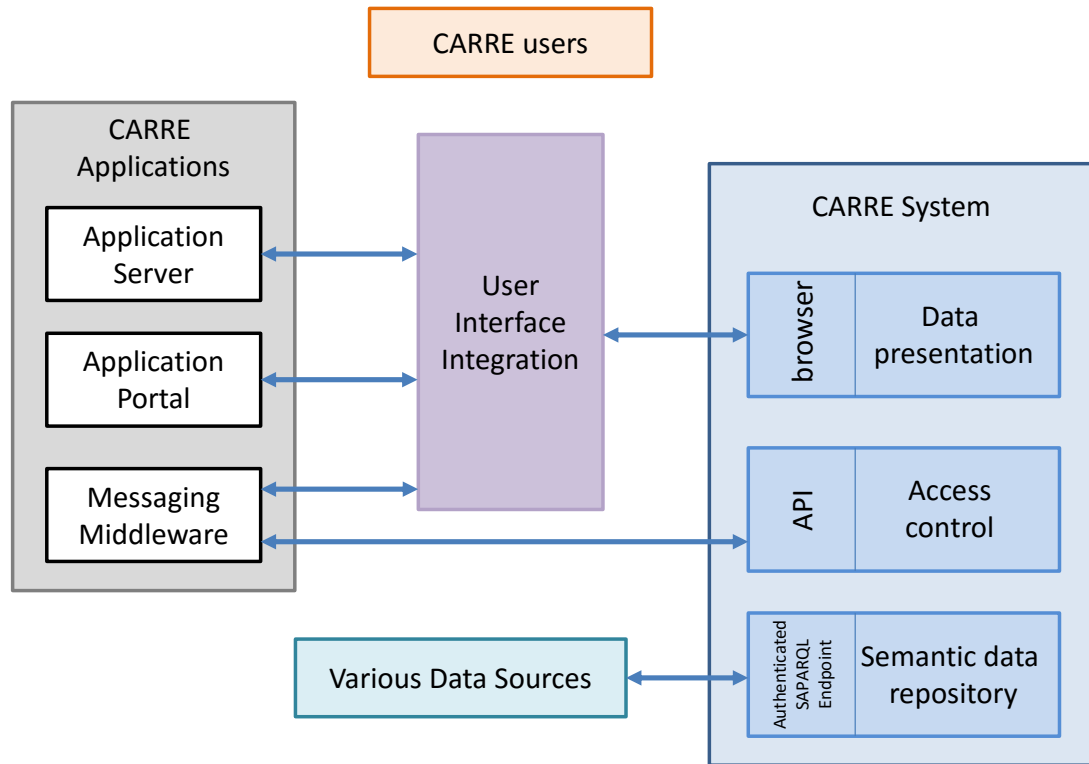
Figure 3. CARRE user interface integration.
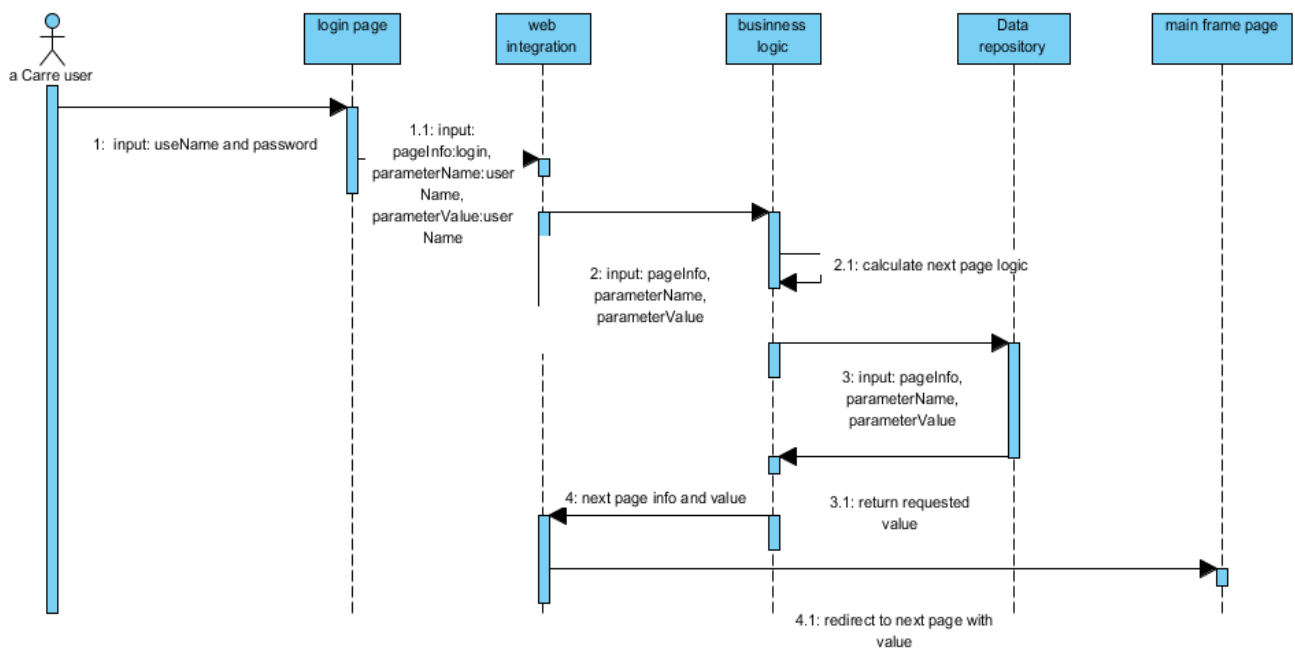
Figure 4. Illustration of the logic of the interface integration.

2.3. High level function diagram of CARRE
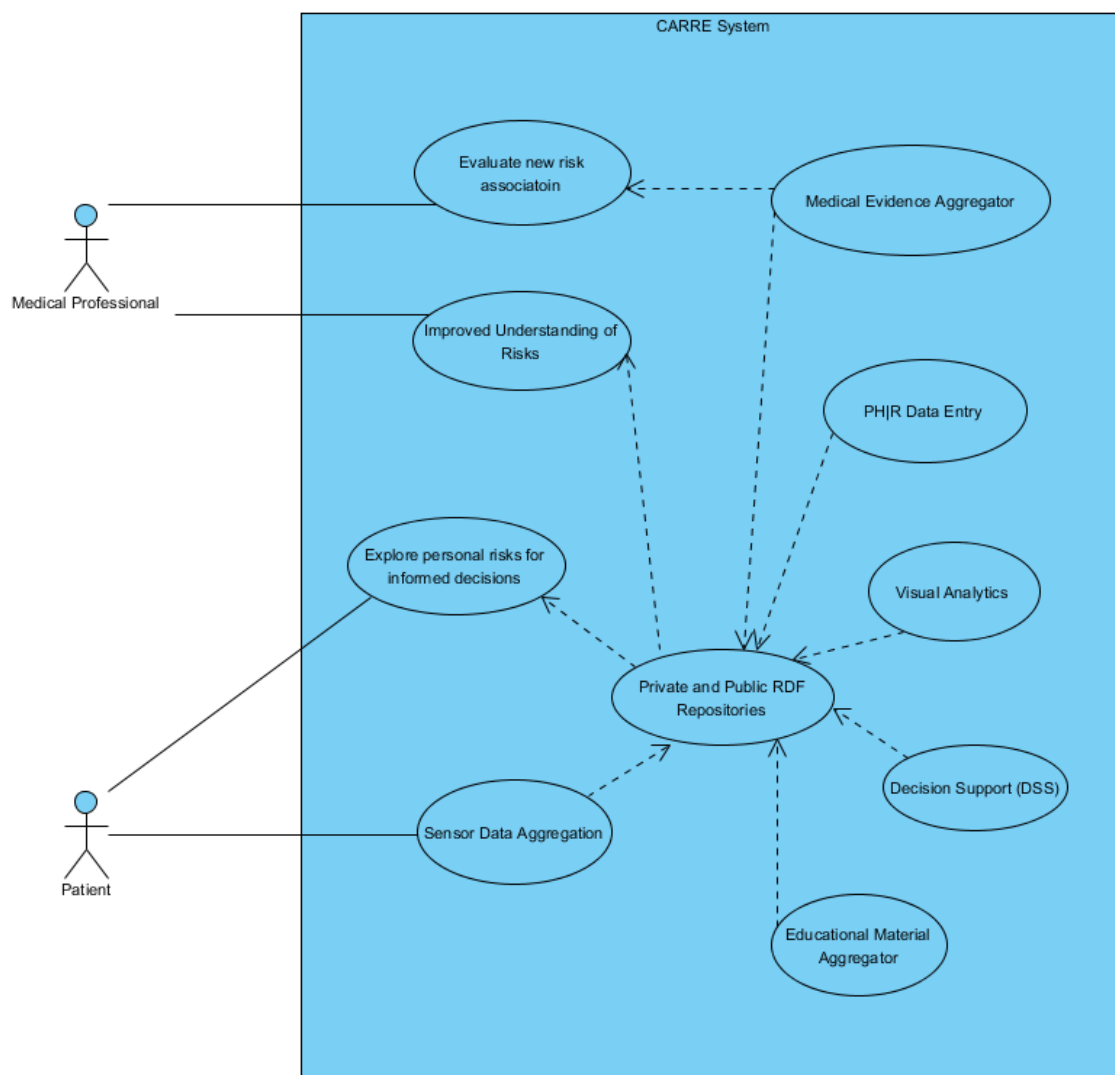
Figure 5. A high level function diagram in CARRE.

D2.1 defined the CARRE use cases in which two types of users were identified:

- Patients, including their carers, and also including general users who are interested in using CARRE services to monitor their potential risk of developing diseases.
- Doctors, including the patients' doctors, or in general, medical professionals or medical researchers. Medical researchers may be interested in medical evidence data aggregation.

Figure 5 presents a high level function diagram in CARRE.

3. Data integration

The CARRE data repository, described in deliverable D.4.1, is designed around the use of RDF, the standard semantic data format used for Linked Data. The ontologies described in D.2.4 constrain this data according to the CARRE information model, enabling the various CARRE components (and, in the case of public data such as risk factor association data, the wider Linked Data world) to integrate smoothly at the data level. The use of Semantic Web standards is essential to this integration: by structuring and annotating data in accordance with

explicit machine-readable specifications of what it means, it is possible to make connections and draw inferences which would not otherwise be possible.

3.1. RDF repository structure

RDF datasets can be structured into graphs, which are named entities able to contain arbitrary numbers of RDF triples that, in essence, give a name to a particular dataset. Certain RDF datastores (in particular, the Virtuoso RDF store used within CARRE) allow for individual graphs to be subject to access control. If a user who does not have permission to read a particular graph tries to query its contents via SPARQL, no results will be returned. This feature is used as the basis for the storage of confidential, patient- or medical-expert-specific, data in CARRE. Where we refer to the "private RDF repository", we mean the CARRE repository containing RDF graphs, where each graph belongs to a particular user (patient or medical expert) and with access controlled by that user. In contrast, the "public RDF repository" contains RDF graphs for which there is no access control for reading data: anyone has permission to read and make use of data in the public repository. There are, of course, access controls in place with regard to the addition or modification of public data, to ensure the integrity of the medical knowledge stored in the CARRE system.

The CARRE private RDF repository contains a separate graph for each user of the CARRE system, whether patient or medical expert. All data pertaining specifically to an individual user is stored in the relevant user graph. CARRE user graphs have names of the form *https://carre.kmi.open.ac.uk/users/USERNAME*, where *USERNAME* is the identifier which the relevant person uses to log in to the CARRE system. There is no requirement that *USERNAME* corresponds to that person's actual name or public identifying feature.

There are three main graphs in the CARRE public repository:

- Risk data graph:
    - *http://carre.kmi.open.ac.uk/public*: contains the latest public, curated, medical evidence aggregated by the project, in RDF according to the relevant CARRE ontology.
    - *http://carre.kmi.open.ac.uk/riskdata*: contains the latest public medical evidence aggregated by the project, in RDF according to the relevant CARRE ontology. Data in this graph has not necessarily yet been reviewed or checked for quality, and should not be used for risk calculation. It may well contain data more up-to-date than that in */public*. Once a risk factor (or modification to a risk factor to account for new evidence) has been subjected to full review and quality assurance, it is copied from the */riskdata* graph into */public*.
- Educational resources graph:
    - *http://carre.kmi.open.ac.uk/edudata*: contains educational materials and ratings of them in RDF as described in D.3.4.
- DSS data graph:
    - *http://carre.kmi.open.ac.uk/dssdata*: Contains DSS algorithms and alerts for patients and medical experts in RDF as described in D.6.2, D.6.3 and D.6.4.

3.2. Component interaction with RDF repositories

Figure 6 illustrates the interaction of CARRE components with a patient graph of the private RDF repository. In terms of data storage, each individual CARRE aggregator (i.e., sensor, PHR, web lifestyle and ECG aggregator) interacts only with the private repository and stores data relating to a particular user in that user's private graph. Accordingly, the PHR manual entry system interacts with user's private graph in order to insert and manage health records that are provided by user. The management system for personal aggregators stores and revokes the user's access tokens of particular manufactures in order to be used by the corresponding aggregators. The decision support service stores alert and recommendation data relating to a particular user and, used in collaboration with the DSS alerts entry system, makes it possible to create, store and manage personalized alert data for each user. Finally, the visual interface and visualisation component visualise and present the user's personal data that has been collected.
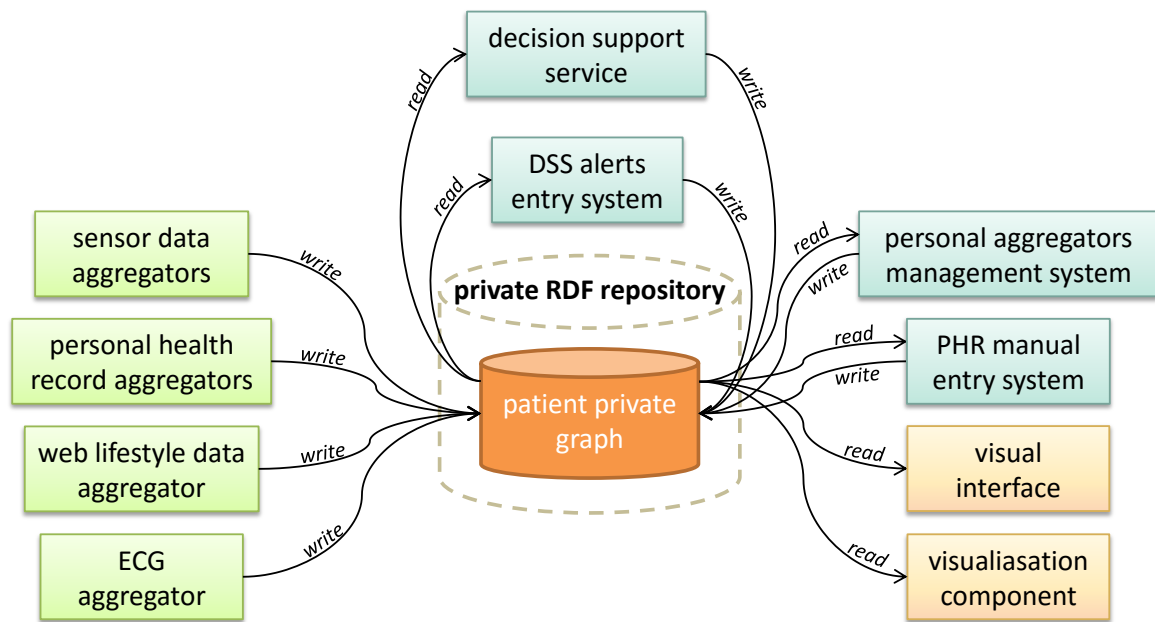
Figure 6. Interaction of components with a patient graph of the private RDF repository.
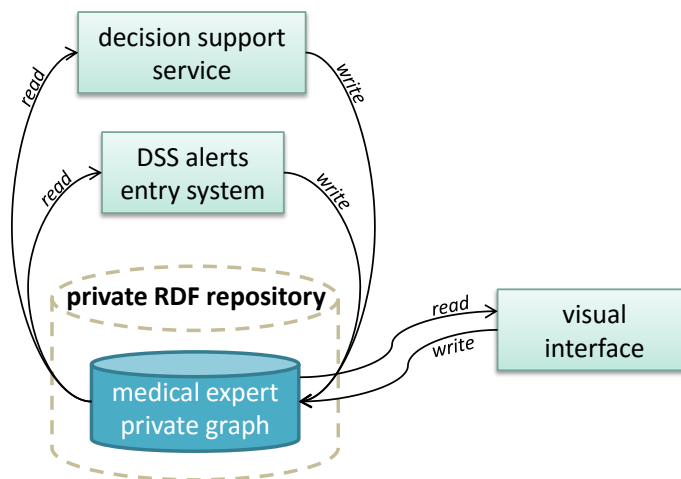
Figure 7. Interaction of components with a medical expert graph of the private RDF repository.

Figure 7 illustrates the interaction of CARRE components with a medical expert graph of the private RDF repository. The decision support service stores alert data relating to particular patients being monitored by the medical expert. The DSS alerts entry system gives the medical expert the option to create, store and manage personalized alert data for him/her and possibly dedicated to a specific patient. The medical expert manages his/her patients and gains access to their personal data through the visual interface.

Figure 8 illustrates the interaction of components with the public RDF repository. The risk factor entry system interacts with the risk data graph in the public repository in order to allow the insertion, editing and review of risk factor data by medical experts. The educational resources aggregator interacts with its own graph, the educational resources graph, and the risk data graph in order to retrieve, store and rate educational materials in its graph. The DSS alerts entry system inserts, edits and manages the DSS algorithms and alerts for patients and medical experts. All of the other components read data from the public RDF repository in order to utilize this knowledge. The visual interface and decision support service component retrieve data from both repositories – generic public data relating to risk factors, educational material and DSS alert messages from the public repository, and patient-specific data from the private repository. These different types of data are integrated according to the requirements of each component. This integration is made possible by the use of the shared CARRE ontologies, vocabularies and RDF data models.
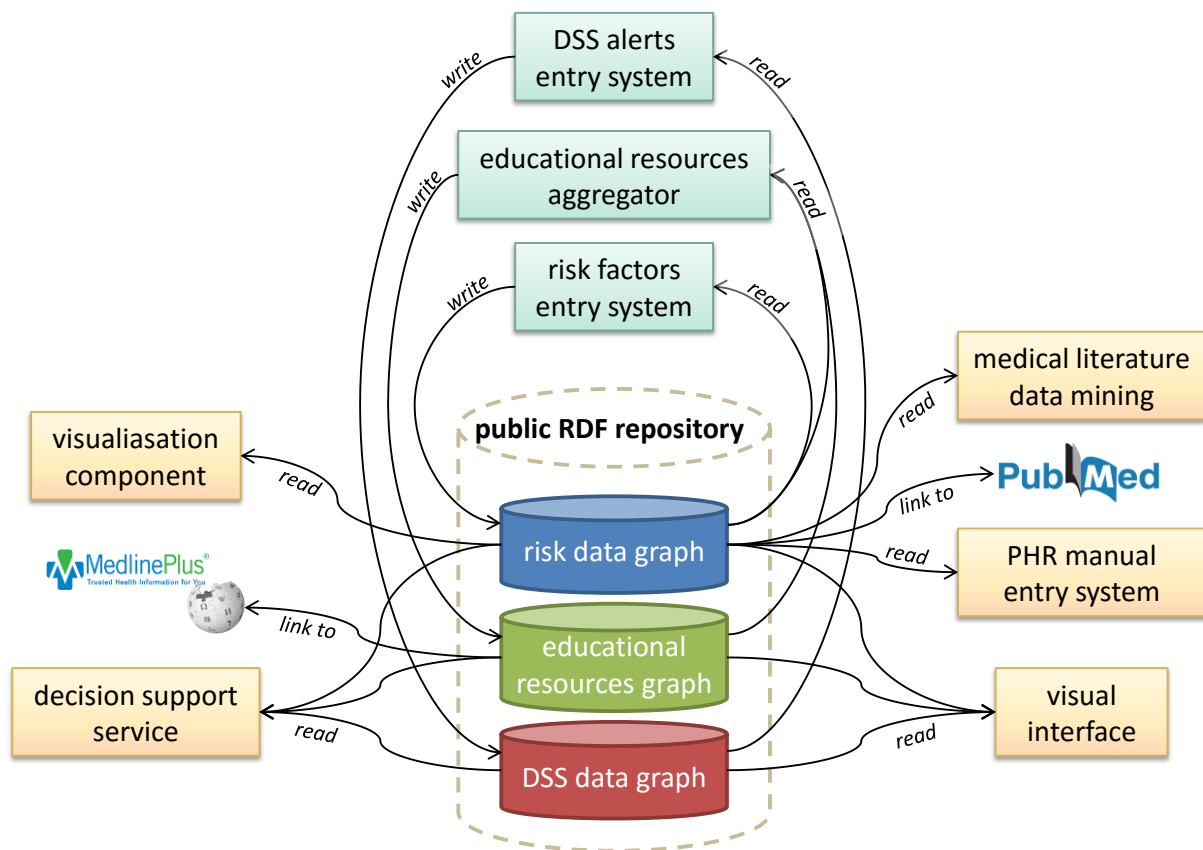
Figure 8. Interaction of components with the public RDF repository.

4. Access control

4.1. Overview

The authentication and access control models for CARRE were outlined in deliverables D.4.1 and D.4.2. Access to an individual user's RDF graph (or write access to one of the public graphs) is granted through use of a token, acquired by going through the standard secure authentication process (based on OAuth); this is shown in Figure 9. Any CARRE component that requires a restricted level of access to any graph will redirect the user to the CARRE repository login page. On successful login and authorisation, the component is provided with a token which can be used to interact with the CARRE RESTful API in order to retrieve, add, or modify data in the relevant graph. This redirection ensures that users are never asked to enter login details into anything other than the secure CARRE repository login page. Tokens can be revoked by the user at any time, at which point they cease to function, ensuring that a user has full control over access to all private data.
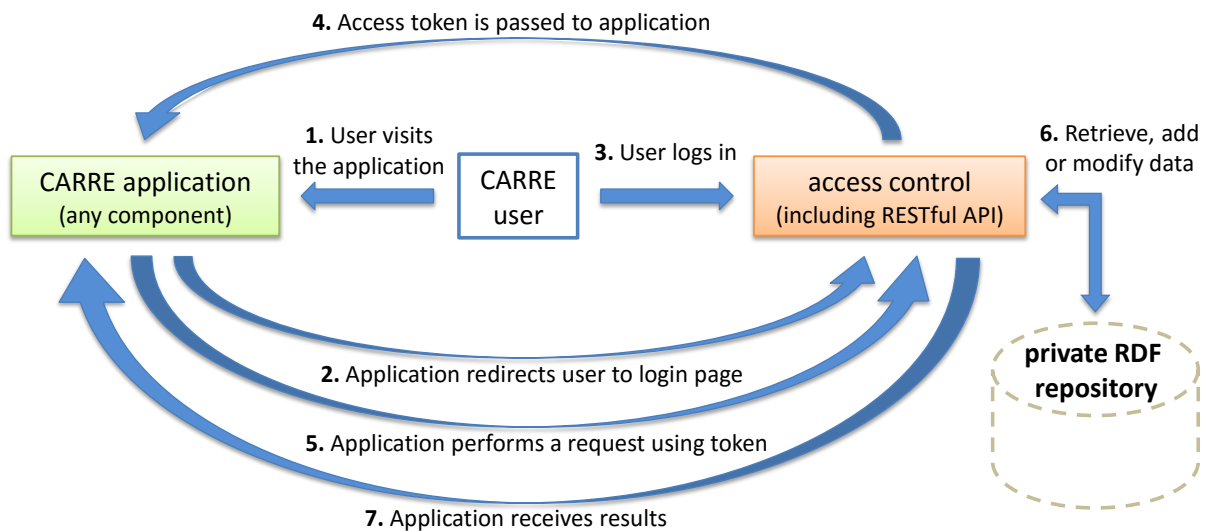
Figure 9. Workflow for login authentication and authorisation.

In order to aid integration – in particular, to support the CARRE tools for medical experts – the concept of a secondary token has been introduced. A patient may wish to grant permission to a particular medical expert (e.g. his/her own physician) to allow access to his/her private data. This is done by selecting that user in the visual interface. A new secondary token is generated, and stored in the private graphs of both the original user and medical expert. The workflow of the medical expert accessing patient data is shown in Figure 10. The medical expert has a limited capability to access that user's data using the secondary token. In particular, it is not possible for the holder of a secondary token to subvert the access control or authentication mechanisms for the user in question. A user may revoke a secondary token at any time; this immediately withdraws access to their private data from the medical expert user.
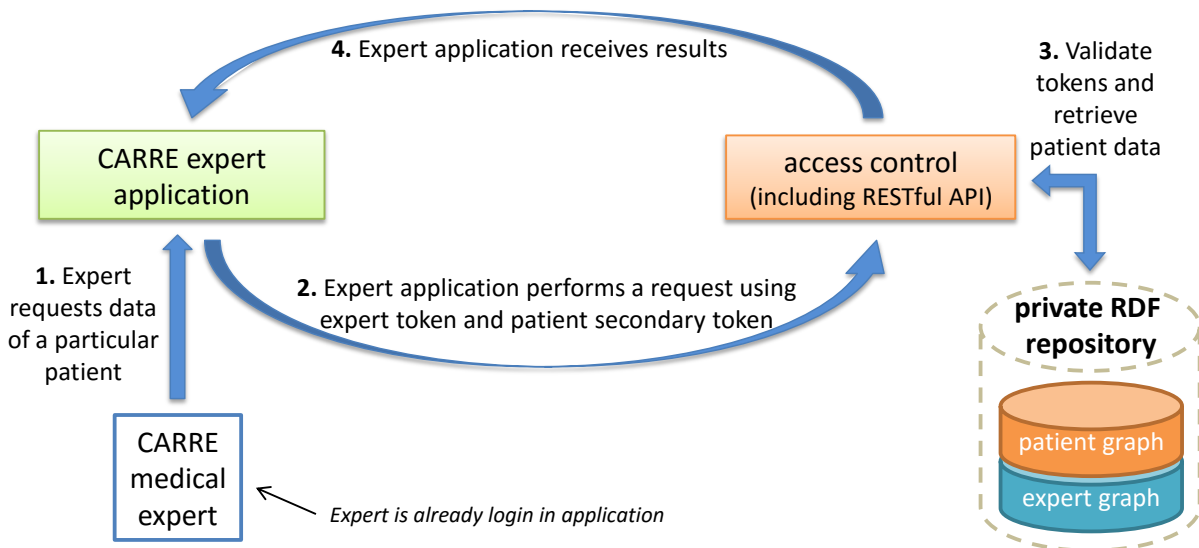
Figure 10. Workflow for a medical expert accessing patient data.

4.2. Technical details on authentication and authorisation

4.2.1. Authentication and authorisation

The authentication strategy implemented for CARRE users is based on external authentication with SSO[4] (Single Sign On) support. The process is described as the following:

1. The user is redirected from a CARRE client application to a secure login interface.
2. The user enters his/her credentials and logs in.
3. The CARRE authentication server validates the credentials and sets a cookie on the user's browser and redirects him back to the source CARRE subdomain which is validated by the authentication server (e.g the user cannot be redirected to *some.evil.com*).
4. The CARRE client application can now make authenticated requests to the CARRE RESTful API.

The main advantages of this authentication strategy are:

- The cookie is set with "Secure" and "Http-Only" flags so it cannot be accessed by the client application (javascript or html) cannot be sent over non-https connections. As a result this method is immune to XSS (Cross-site Scripting) attacks[5].
- Cookies are automatically processed by web browsers and web servers which provides a good integrated experience among all the CARRE applications.

Accordigly, the limitations of this are:

- This method is vulnerable to CSRF (Cross-Site Request Forgery) attacks[6] which exploit the trust that a site has in a user's browser. This can be prevented by including a secondary token to the request validating that this request is performed by a validated application.
- CARRE applications must be under the same root domain (e.g. entry.carre-project.eu, visual.carre-project.eu etc.) in order for the cookies implementation to work.

4.2.2. Further security improvements

The CARRE authorisation process is based on OAuth 1.0 which is an API-based authorisation protocol that allows a third-party website or application to authorize access to a user's data without requiring user's login credentials. The industry standards on security and authentication techniques are evolving rapidly as the Internet continues to expand. As a result, the OAuth 1.0 will be replaced by the next version OAuth 2.0 which introduces the concept of a refresh token and limited duration of the access token. The refresh token is used to request new access tokens without compromising the user experience, e.g without requesting the user to re-login.

Another future security upgrade will be in the authentication process. As described above, the CARRE authentication is currently based on cookies, which has its advantages and its limitations. The Internet is gradually evolving towards token-based authentication and this can be implemented using JWT[7] (JSON web tokens). This new approach will replace the storage method and introduce a client encryption technique for saving the user state in the client and not in the server, as with the cookies approach.

---

[4] https://en.wikipedia.org/wiki/Single_sign-on

[5] https://en.wikipedia.org/wiki/Cross-site_scripting

[6] https://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF)

[7] https://jwt.io/

5. User interface integration

CARRE applications are integrated via the user interface, which is a data presentation layer and a visual interface for users to access the existing data, and add new data to the CARRE data repositories.

5.1. Overview

From a system architecture point of view, CARRE is composed of multiple subsystems or modules. The underlying infrastructural subsystems include the public and private repositories and the access control. The middle layer subsystems include the DSS system, sensor aggregator, medical evidence aggregator and educational material aggregator. The surface layer subsystems which directly interact with the user include the PHR entry system, risk entry system, and the visual analytics module.

However, the data presentation layer itself is not a standalone subsystem but is composed of a set of loosely coupled components, including the main visual interface and visual analytics, user interfaces from several functional subsystems/modules and the supporting infrastructural components.

Among the related components, the main visual interface and the visual analytics module are the core components which totally belong to the data presentation layer. The risk entry system, PHR entry system, DSS system, medical evidence aggregator, etc. are categorized as functional modules whose user interfaces lie in the data presentation layer. The public private data repositories and the access control subsystems communicate with other components of the data presentation layer and are categorized into the infrastructural components for the data presentation layer.

This section provides a brief introduction to components of the data presentation layer, followed by details of integration from three levels: component, infrastructural and application.

5.2. CARRE components

5.2.1. Core components of visual analytics

5.2.1.1. The main visual interface

The main visual interface (Figure 11) provides user authentication and a web-based framework for directly hosting other components or for linking to front-ends of other related components. It hosts visual analysis components and the medical evidence aggregator. The web-based main visual interface framework is composed of a dashboard, visual analytics component containers and links to other related components. The main visual interface also provides the entry point for users to login to the CARRE system with the OU's authentication mechanism, thereby achieving secure user authentication and access control.
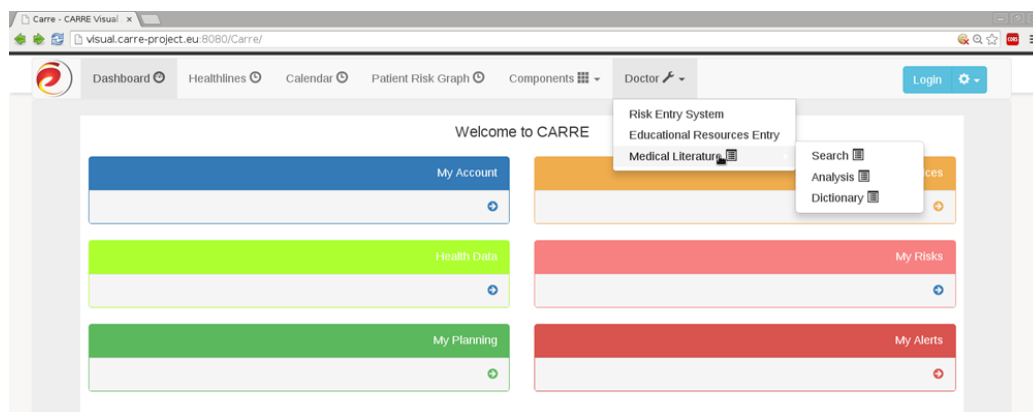
Figure 11. Screen shot of the main visual interface

5.2.1.2.   The visual analytics component

The visual analytics (VA) component combines automated analysis techniques with interactive visualisations for effective understanding, reasoning and decision making. In the CARRE system architecture, the data sources of the VA component can generally be categorised into sensor data collected by sensor data aggregators, PHR data collected by PHR aggregators, personal risk data calculated by the DSS component, and the public risk associations extracted from the medical literature. The private data associated with individual patients are stored in the private RDF repository and protected by the access control component. Users need individual credentials to gain access to the private data they are authorised to access. The public risk associations which represent the medical knowledge of risks are stored in the public repository; users do not need individual credentials to access the public data.

5.2.2.   Functional components

Other related functional subsystems which have their own user interface components and are thus related to the data presentation layer include:

- Risk Entry system
- PHR entry system
- DSS system
- Medical Evidence Aggregator

The user interfaces of these are integrated into the main visual interface either by links or by direct integration.

5.2.3.   Infrastructural components

Infrastructural components of the data presentation layer are CARRE components which provide system level infrastructural support (such as access control, data storage and data access, etc.) to the data presentation layer as well as other modules.

Infrastructural components of the data presentation layer include the public repository, the private repository subsystem and the access control subsystem.

5.2.4.   Role of data presentation in CARRE

The visual data presentation relies on the public RDF repository and the private RDF repository to provide data of risk associations as well as data of personal health. While accessing the private data, the visual data presentation components depend upon the access control component to provide secure access the data.

The relationships among the main visual interface, the visual analytics component and other related components of the data presentation layer are shown in the following component diagram (Figure 12). For clarity, we show only the dependency of the main visual interface on the access control module, but all other components also depend upon the access control to provide user authentication, user management and secure resource access control.
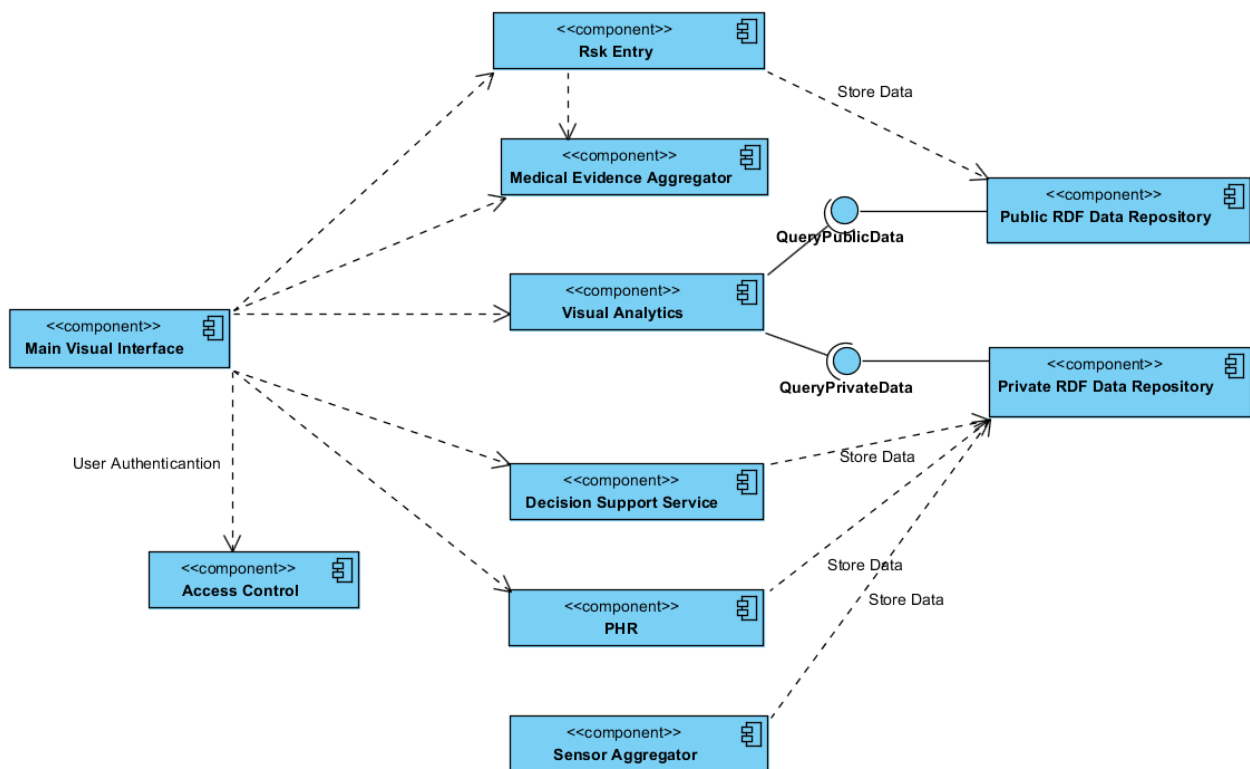
Figure 12. Component diagram shows the relationship of the main visual interface, visual analytics and other related components

5.3. Component level integration

5.3.1. The main visual interface

The role of the visual interface is to provide links or container for the user interfaces of other related components, such as the medical evidence aggregator, the risk entry system, the PHR system, the DSS component, etc.

The main visual interface provides user authentication for an individual user. It also provides a summary of health information and a user profile display.

5.3.1.1. Dependency

The main visual interface is directly dependent upon the access control module to provide user access control. It provides the user login interface to gain access to the system. The user credentials are stored in the local cookie which is used by other components to retrieve the data stored in the private RDF data repository. It also needs the private repository to read the user's profile.

The main visual interface also depends on the VA component to provide visual analysis of the public and private data.

5.3.1.2. Communication

The main visual interface communicates with the access control module and the private repository.

5.3.2. Visual analytics

The role of visual analytics is to visualise sensor data, PHR data, medical data, risk factor data and the integrated visual analysis of health data and risk factor data. As a component focusing on visual information presentation,

VA is closely integrated to the visual interface as container. It requests data from the public and private repository, processes and analyses the data, and presents the results to the user visually on the display.

5.3.2.1. Dependency

The visual analytics component is dependent upon the main visual interface for access to the user's profiles and credentials so as to gain access to the private repository.

The visual analytics component relies heavily on the public and private repositories to retrieve the public and private data. It uses RESTful APIs as well as SPARQL queries to access the data from them. For the public repository, the query can be patient-specific risk associations. For the private repository, the data can be the user profile, sensor data, PHR data and the DSS data. Each of these can be accessed via the unified measurement API provided by the private RDF data repository.

In addition, the visual analytics component indirectly relies on the decision support service to calculate the risks of diseases based on the medical evidence provided and the user's health data. The risk data are calculated, then stored in the private RDF repository by the decision support service for retrieval by the visual analytics component and other components.

The visual analytics component is embedded in the visual interface as client-side web components and is invoked by user actions.

5.3.2.2. Communication

The visual analytics component communicates with the public and private repositories via RESTful APIs and SPARQL queries.

5.4. Infrastructure level integration

5.4.1. User authentication

User authentication is achieved via the access control module. The visual interface provides the login function. When the user clicks on the "Login" button in the visual interface, the web page automatically redirects to the CARRE user authentication website. If the user inputs the correct user name and password, the user's credentials will be stored in local cookies.

The user authentication URL is

 https://devices.carre-project.eu/devices/accounts/login?next=FROM_URL

The FROM_URL is the caller's URL. On a successful login, the page will redirect back to that URL.

5.4.1.1. Patient user authentication

For the main visual interface, the user authentication is called by the following URL https://devices.carre-project.eu/devices/accounts/login?next=http://visual.carre-project.eu. This is shown in Figure 13.
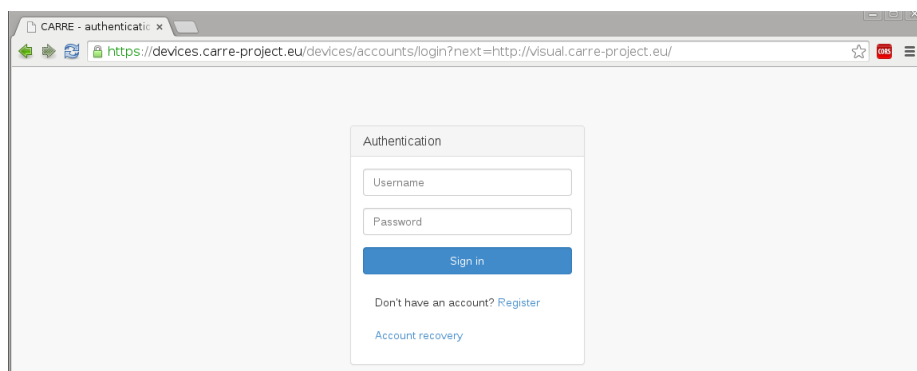
Figure 13. User authentication page

After a successful login, the user's name will be displayed on the main visual interface.

5.4.1.2. Doctor's user authentication

Doctors have to check the health data of their patients, so they need not only their own credentials, but also those of their patients. The CARRE access control module provides doctors with a "secondary token", as described earlier, to allow limited access to the patients' private data.

5.4.2. Data retrieval and access

5.4.2.1. Data storage

In CARRE, the data storage is achieved via the CARRE data repository subsystem which acts as the central point of information storage for all CARRE applications. It conforms to the principles of the Semantic Web and the guidelines of Linked Data. Information stored in the CARRE repository consists of "RDF triples". RDF (Resource Description Framework) is a standard format for representing semantic data on the Web; an item of RDF data is a triple, which corresponds to a statement of the form "subject predicate object". Each term is a URI, often drawn from a standard vocabulary or ontology, making it easy to link triples from different sources – that is, to allow Linked Data. RDF can be accessed through a SPARQL[8] endpoint: SPARQL is a query language, much like SQL in syntax. The triples stored in the CARRE repository are either public or private. For private data, data privacy and security mechanisms have been deployed via the access control module.

5.4.2.2. The Public RDF repository

The public RDF repository stores general medical knowledge relating to risk associations, evidence and observables, and is available for public querying without authentication, as it contains no personally identifying data for any patients and serves as a general-purpose resource for medical knowledge in a semantic format.

5.4.2.3. Private RDF repository

The private RDF repository stores data relating to individual patients, in a secure and access-limited fashion, with each patient's data in a separate and restricted RDF graph. All data from personal sensors, personal health records and any decision support service recommendations for an individual are stored and queried from that individual's private graph.

5.4.2.4. Retrieving data

The RESTful API method for accessing CARRE data has been described in detail in deliverables D.4.1 and D.4.2, including the privacy-by-design features for private data. An authenticated and approved application can retrieve an access token per user from the API, which is used to authenticate all calls which access that user's private RDF graph. API calls relating to public data may be performed without any token.

The following are example API queries used by the visual analytics component:

- *Retrieving the Measurement List*

    The list of the measurements can be acquired from the following API:

    https://devices.carre-project.eu/ws/measurementsList?token=TOKEN

    where TOKEN is the user token. The query returns an array of URIs of the measurement variables. The following URI represents the step measurement name:

    http://carre.kmi.open.ac.uk/ontology/sensors.owl#has_steps

- *Retrieving Measurement Data*

    Most of the user data reading is based on the OU measurement API:

    https://devices.carre-project.eu/ws/measurement?name=MEASUREMENT_NAME&token=TOKEN

---

8   https://www.w3.org/TR/rdf-sparql-query/

where MEASUREMENT_NAME is the RDF variable name and token is the user token which is acquired from the CARRE user management service.

- *Retrieving Risk Elements*

  Retrieval of risk elements can be achieved by the following API query:

  https://devices.carre-project.eu/ws/instances?type=risk_element&language=LANGUAGE

  It returns an array of RDF triples which have risk elements as the subject.

- *Retrieving Risk Factors*

  Retrieval of risk factors names can be achieved by the following API query:

  https://devices.carre-project.eu/ws/instances?type=risk_factor

  It returns an array of RDF triples which have risk factors as the subject.

SPARQL is the standard query language for RDF data. The API allows arbitrary SPARQL queries to be submitted via an authenticated call, and also contains several helper methods which wrap specific common queries in an optimised way (for example, retrieving a daily summary of sensor readings for a user). Compared to the API method, it is more flexible and customizable.

The following are example SPARQL API queries used by the visual analytics component:

- *Retrieving the Measurement Data*

```
SELECT DISTINCT ?measurementtype ?ob_name
FROM <https://carre.kmi.open.ac.uk/users/user.name>
FROM <http://carre.kmi.open.ac.uk/riskdata> WHERE
    ?measurement ?measurementtype ?object .
:has_blood_pressure_systolic :has_weight :has_has_blood_glucose_fasting}
   OPTIONAL {?ob a risk:observable; risk:has_external_predicate
?measurementtype; risk:has_observable_name ?ob_name.  }
    FILTER(langMatches(lang(?ob_name), carre_language ) )
```

where **carre_language** is the human language used by the system.

- *Retrieving the Patient Risk Data*

```
SELECT DISTINCT ?risk_evidence ?condition ?confidence_interval_min
?confidence_interval_max ?risk_evidence_ratio_value
?risk_evidence_ratio_type ?risk_factor ?has_risk_factor_source
?has_risk_factor_target ?has_risk_factor_association_type FROM
<http://carre.kmi.open.ac.uk/riskdata> WHERE {
   ?risk_evidence a risk:risk_evidence ;
   risk:has_risk_factor ?risk_factor;
   risk:has_risk_evidence_ratio_type ?risk_evidence_ratio_type;
   risk:has_risk_evidence_ratio_value ?risk_evidence_ratio_value;
   risk:has_confidence_interval_max ?confidence_interval_max;
   risk:has_confidence_interval_min ?confidence_interval_min;
   risk:has_risk_evidence_observable ?ob ;
   risk:has_observable_condition ?condition .
   ?risk_factor risk:has_risk_factor_association_type
?has_risk_factor_association_type;
   risk:has_risk_factor_source ?has_risk_factor_source;
   risk:has_risk_factor_target ?has_risk_factor_target.
   {
      SELECT ?ob FROM <http://carre.kmi.open.ac.uk/riskdata> WHERE {
      ?ob a risk:observable ;
         risk:has_external_predicate ?p.
      }
   }
}
```

5.5. Application level integration

For patients, CARRE is a platform for health risk monitoring and analysis, and for doctors, it is a platform for medical risk knowledge discovery, record and application. In this section, interactions between visual interface components are introduced from the viewpoints of the application and functions. In the following subsections, the components involved in the patient application and the doctor's application are introduced to demonstrate how different components are integrated to provide a unified user interface and system functionalities.

5.5.1. Integrated user interface for patients' application

5.5.1.1. The main visual interface

The main visual interface for patients is shown in Figure 14 which includes a dashboard for information summary, a calendar for event display and visual analytics tools such as the healthlines. It also includes menus for the PHR entry system and the CARRE devices system.
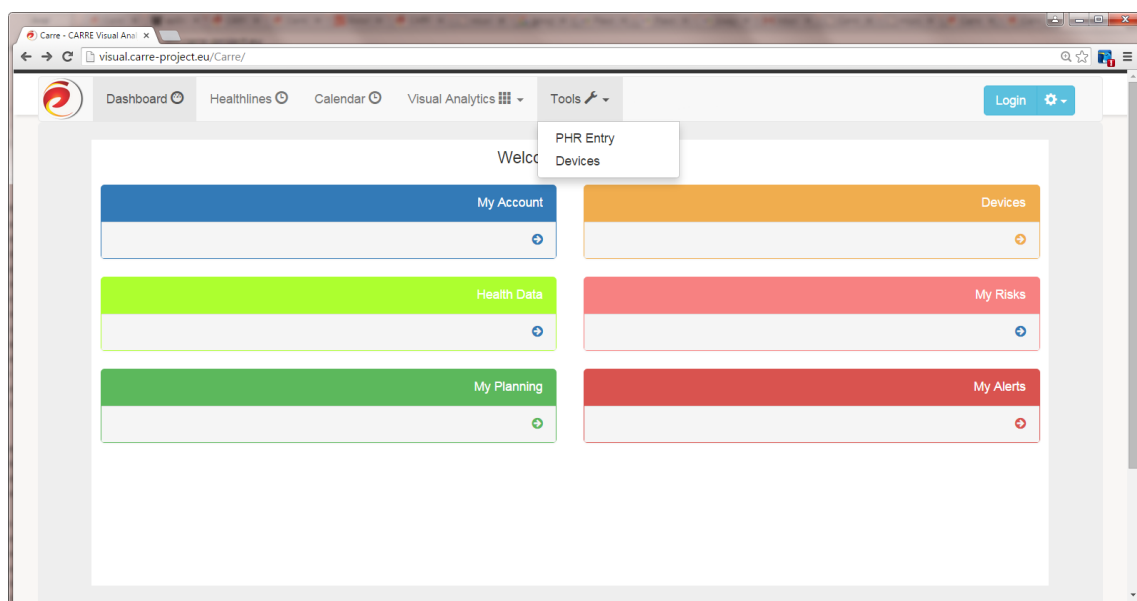
Figure 14. The main visual interface

5.5.1.2. Personal health information summary

Multiple data presentation components can be accessed by the user from the web-based CARRE visual interface. However, as there is a wide variety of data sources and data types, it is difficult for a user to gain an overview of the important features from the scattered health status visualisation. To present the user with a quick overview of their health status, the CARRE visual interface provides a dashboard as a front page. The dashboard provides a summary of the user's latest health status as well as important notifications, as shown in Figure 15. It may include simple visualisation components to present data in a relatively recent period. The following figure shows the example dashboard with data tiles and links to more detailed information.
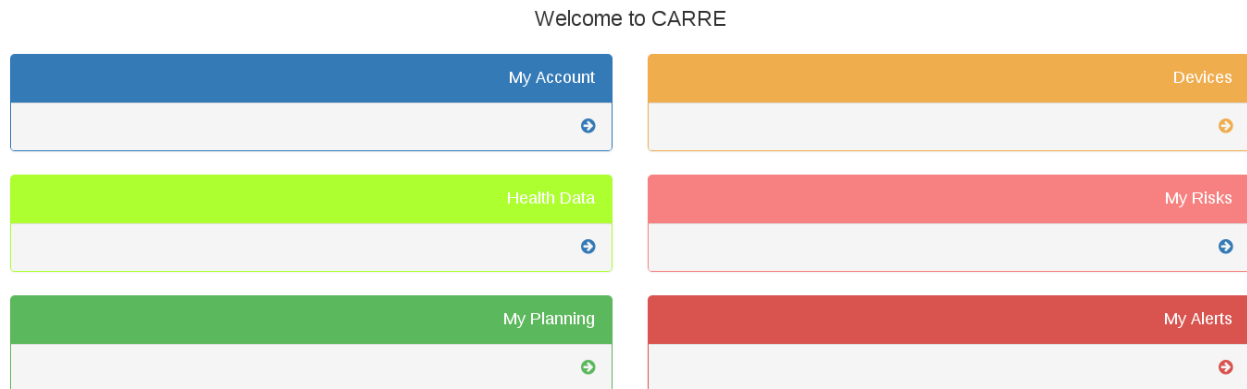
Figure 15. The CARRE dashboard

5.5.1.3. Personal health visual analytics

The visual analytics components introduced in deliverables D5.1 and D5.3 are integrated in the main visual interface as containments as shown in Figure 16. The major visual analytics components include healthlines, a calendar and a risk factor graph. They are integrated and embedded in the visual interface as HTML iframes in the visual interface layout. Separate menus are added for visual analytics components so that the user can easily invoke the visual analytics functions. The interactions between the visual analytics module, visual interface, access control, and public and private repositories in the patient application are shown in the sequence diagram in Figure 17.
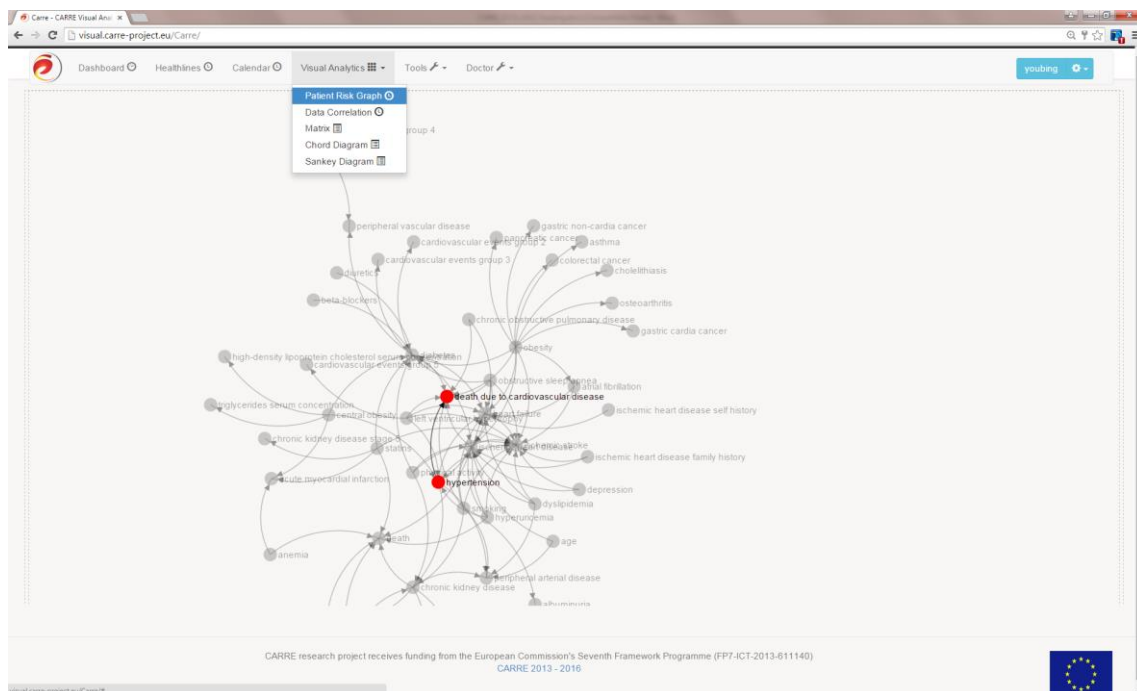
Figure 16. Integration of visual analytics components into the main visual interface
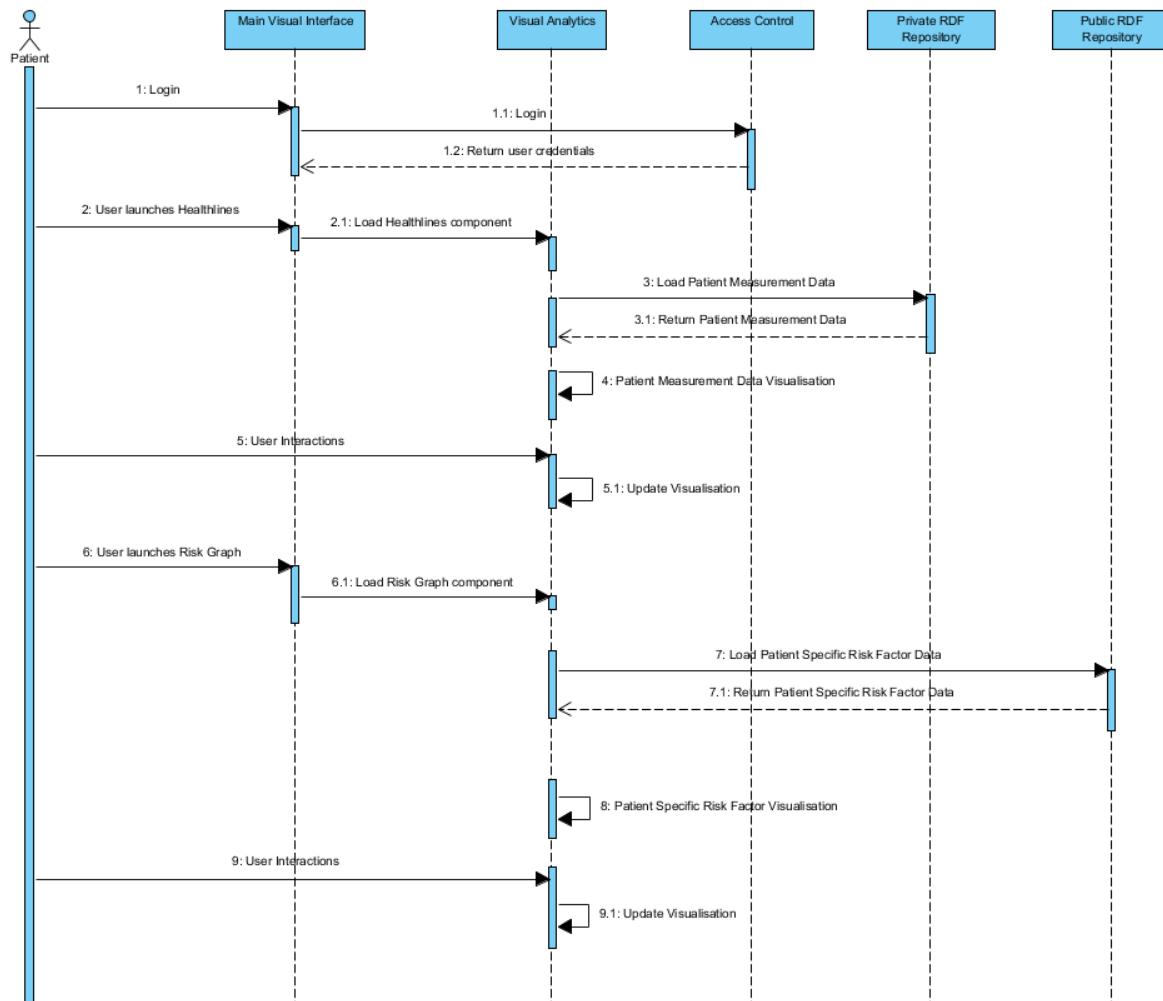
Figure 17. Sequence diagram of interactions of components in the data presentation layer for the patient's application

5.5.1.4. PHR data entry

The patient's health data can be managed using the PHR data entry system. Data management is performed by updating observable measurements in the private RDF repository. Observables that are monitored are taken from public RDF and measurements are saved into the private RDF.

Though designed and implemented as a subsystem, the PHR entry system can be embedded in the main visual interface as an iframe and will share the same user authentication as the main visual interface, as shown in Figure 18.

The interactions between the patient, the PHR entry system, the public RDF repository and the private RDF repository are shown in Figure 19.

Observables are organized into different groups to make it easier for patients to edit and navigate, as shown in the index page of the PHR entry system in Figure 20.
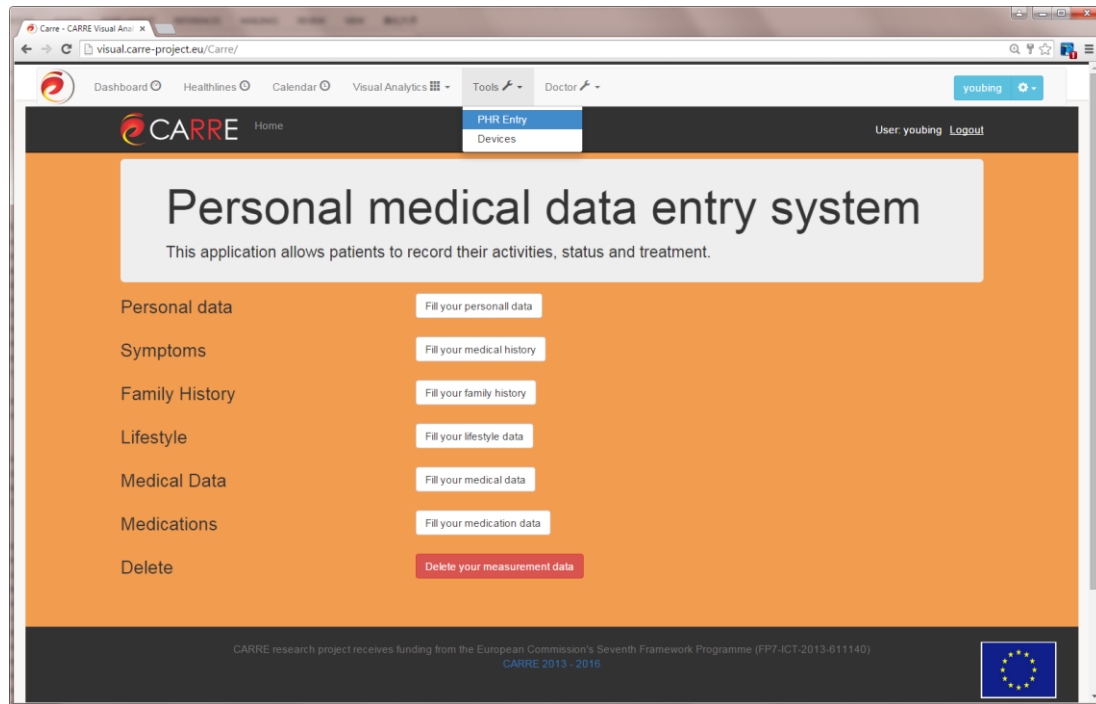
Figure 18. PHR entry system embedded in the main visual interface
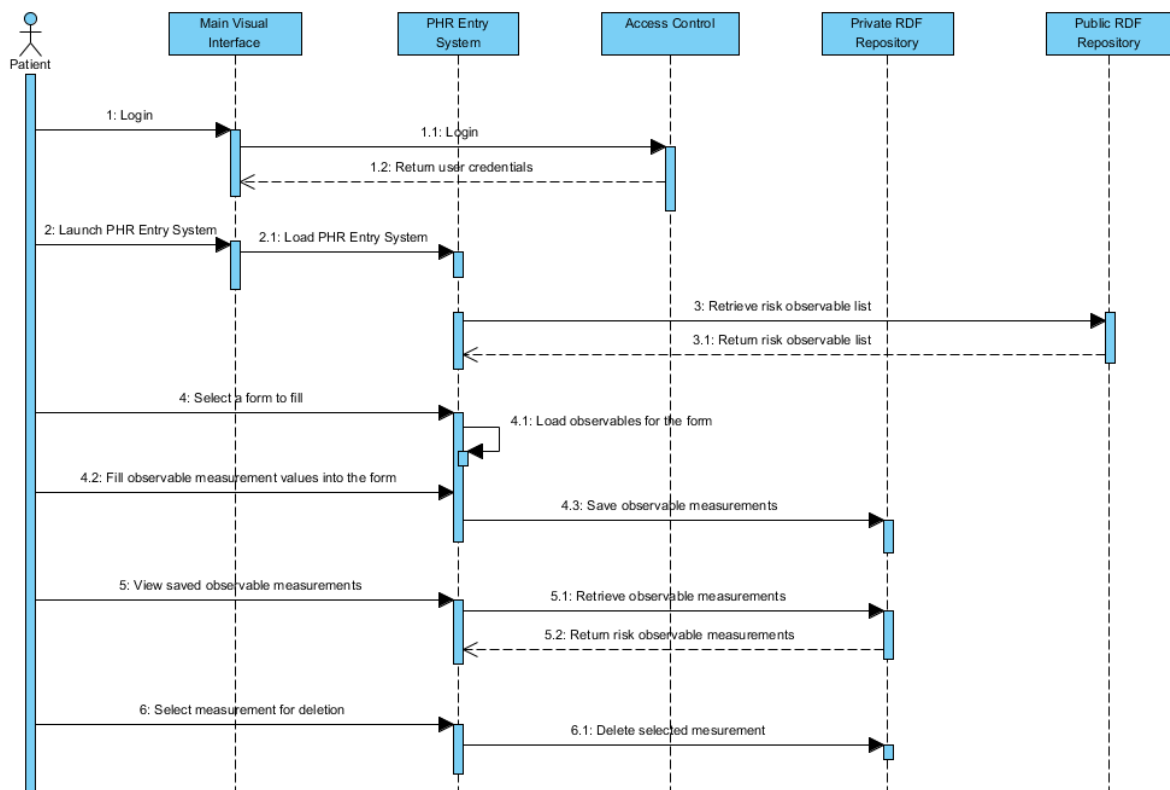
Figure 19. Sequence diagram of interactions between the patient, PHR entry system and the public and private RDF repository
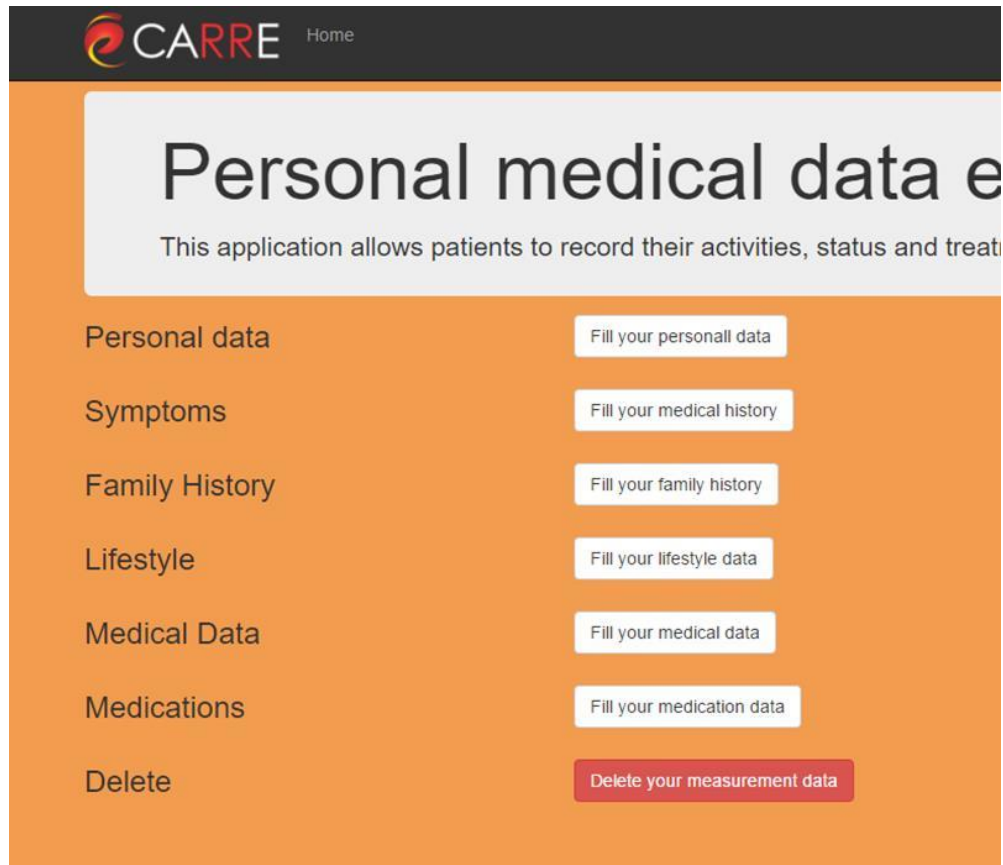
Figure 20. Data are organized into groups in the PHR entry system

A patient can enter new observable measurement by specifying its value and the date of the measurement (date is an optional field, if it is left blank, the current date will be used as measurement date). Observables value can have several forms: either enum value (value from predefined list of options), Boolean value (true or false), integer or decimal number input, as shown in Figure 21.

Figure 21. Filling of observable measurements in PHR entry system

A patient's measurement can be deleted individually, as shown in Figure 22, or as a whole by choosing the "Delete measurements" option in the index page.

Figure 22. Overview of saved observable measurements in the PHR entry system

5.5.1.5. Wearable sensor data collection

Patients can use the CARRE Device system to collect and use data (blood pressure, weight, glucose, physical activity) from dataclouds of selected commercial personal health devices. WEB entry to CARRE Devices is secured by using the same common Login credentials as to the other modules of the CARRE system. Figure 23 shows the entry user interface of the CARRE device system.
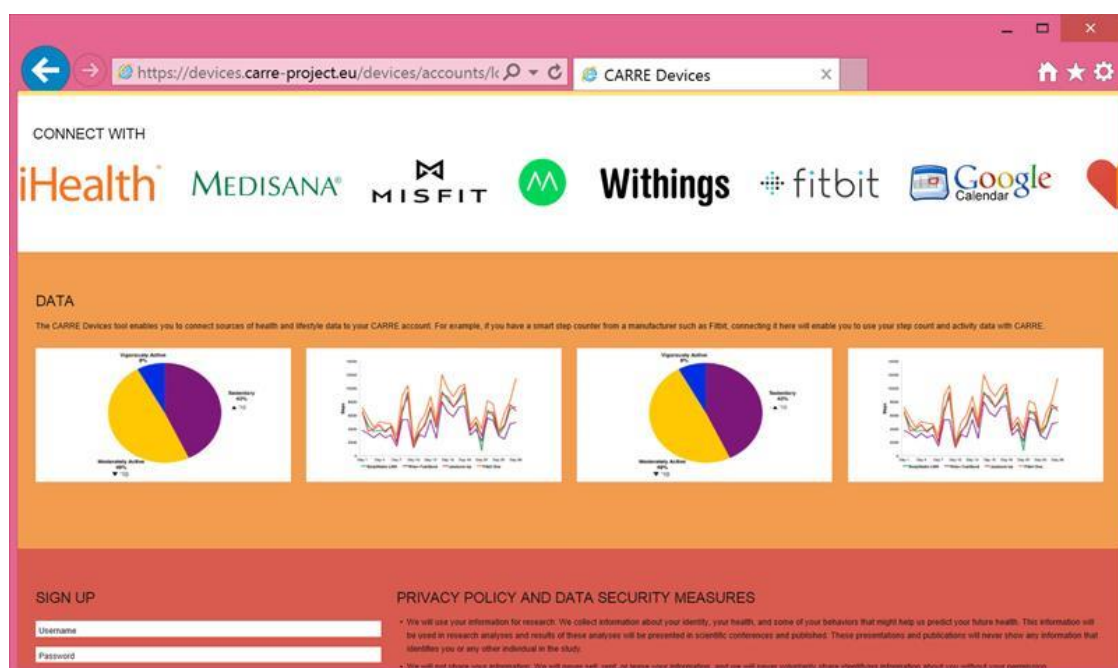
Figure 23. WEB entry to CARRE Devices system

Most commercial devices have similar procedures for allowing data to be sent to CARRE Device services (sequence diagram Figure 24). Connection involves several intuitive steps: selection of commercial vendor, login to cloud connection API and granting permission to access data for the CARRE Devices. From that point, data upload to the private RDF repository is automatic and requires no additional effort from the user.
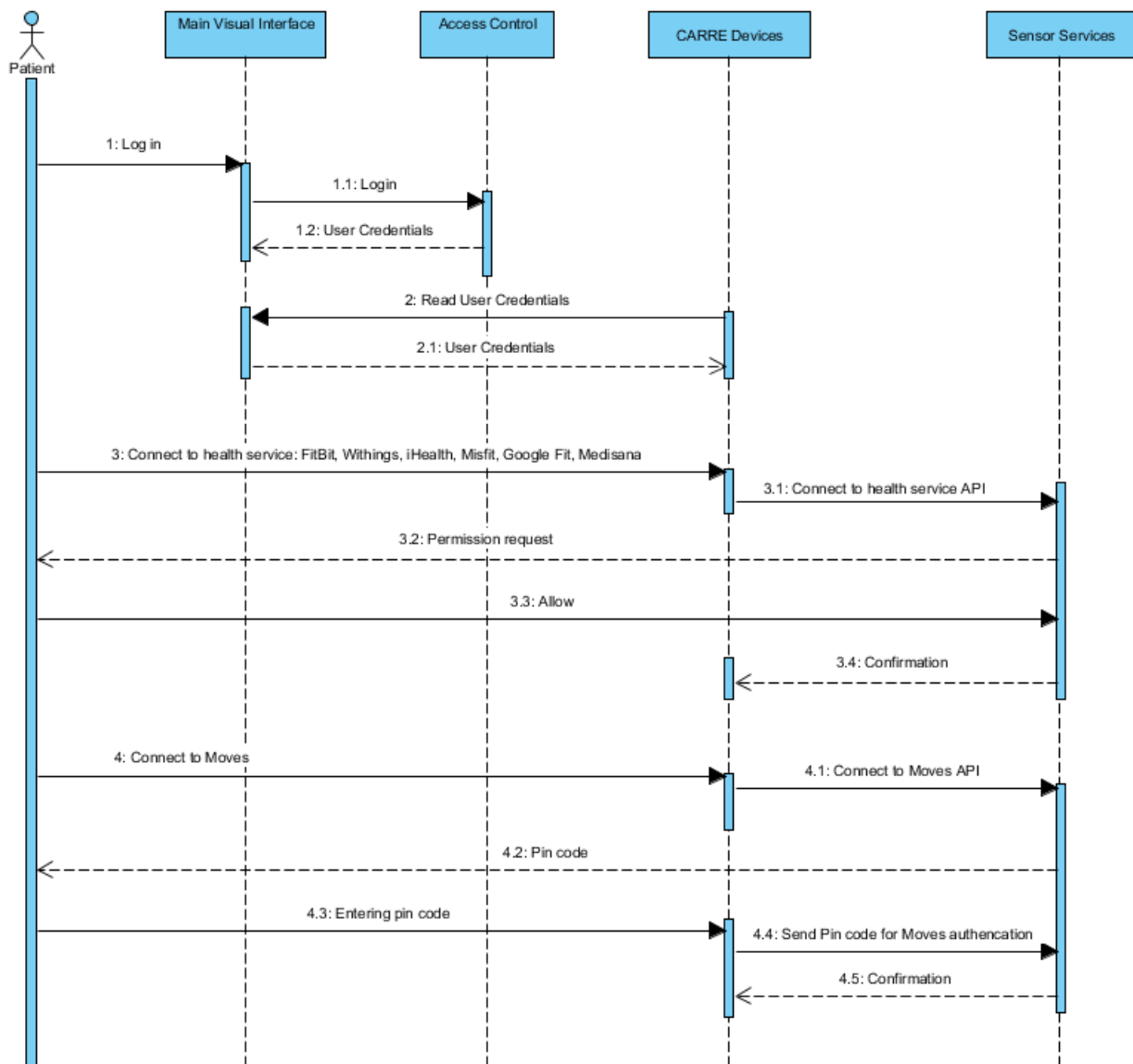
Figure 24. Sequence diagram explaining the process of connecting commercial wearable sensor dataclouds to the CARRE Device service

Figure 25 presents the CARRE devices page when integrated with the main user interface. Figure 26 shows the dialog boxes appearing in the user interface after the user presses the plus sign with the meaning "Connect to a new wearable sensor service". Figure 26.a presents the available commercial sensor providers for selection while Figure 26.b shows an example of choosing one provider (Fitbit) to establish a long-term connection. The user is able to select the data types from the commercial sensor provider to be stored into the CARRE private repository.
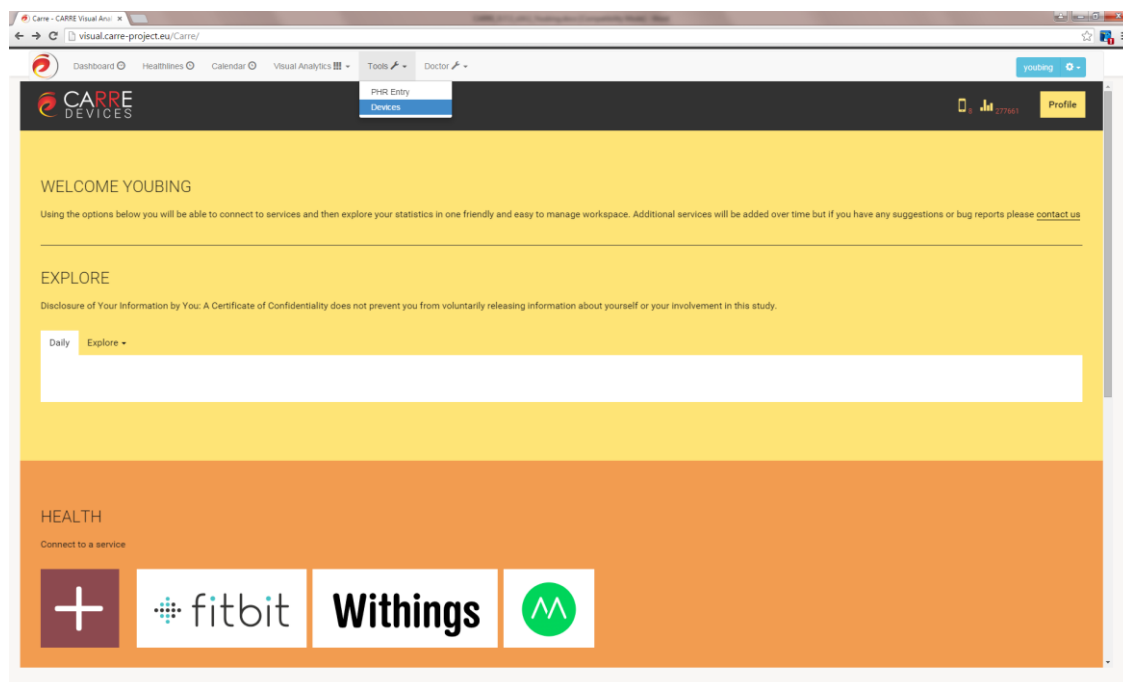
Figure 25. CARRE Devices web page after a successful login

(a)
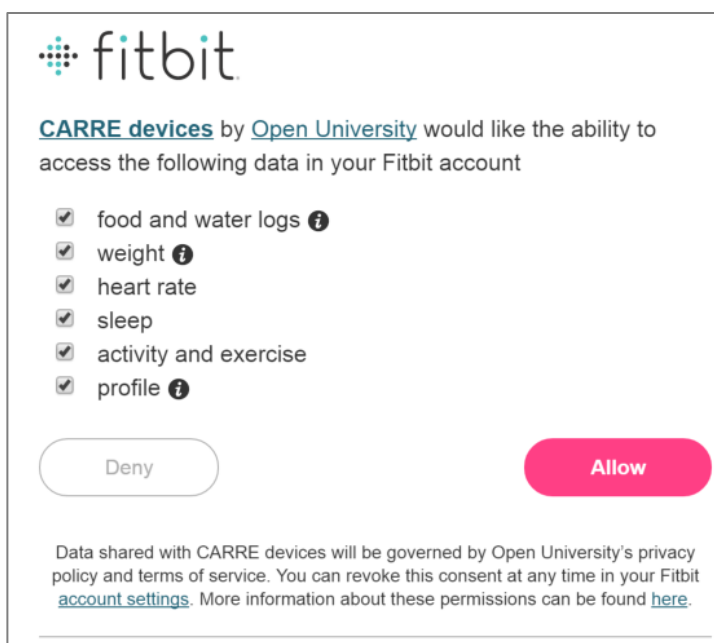
(b)

Figure 26. Screenshots of the user interfaces during connection of commercial dataclouds to CARRE Device services: (a) available commercial data providers for selection, (b) example of Fitbit API response to the request from CARRE Devices

A different type of aggregator is the Web liferstyle data aggregator and ECG aggregator. The goal of Web liferstyle data aggregator is to use patient's web searches in order to extract his/her intention and upload to private RDF repository only the intentions that are relevand to CARRE. Accordigly, ECG aggregator is supposed to accept an .edf file from Emotion Faros ECG (Mega Electronics Ltd, Finland) recorder, analyse it locally with the proprietary signal processing algorithm for detection of brief episodes of heart arrhythmia, send extracted parameters to CARRE private repository and compose analysis report locally for the user. Figure 27 illustrates the interactions of Web liferstyle data aggregator and ECG aggregator with the other CARRE subsystems. The ECG aggregator is integrated with CARRE Web lifestyle data aggregator. It is an installable program working as a background service on the user's computer.
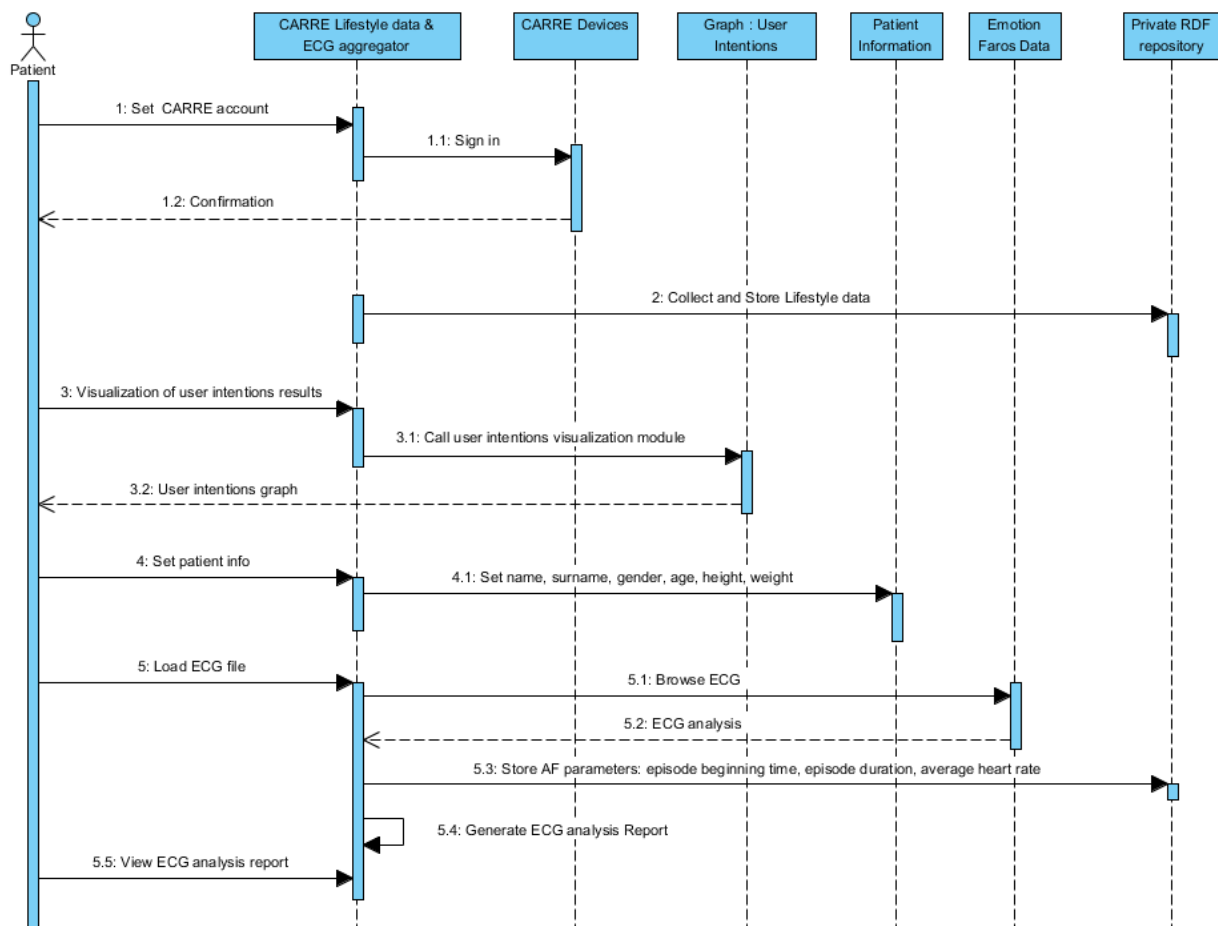
Figure 27. Sequence diagram of Web liferstyle data aggregator and ECG aggregator

Figure 28 shows menus related to the Web liferstyle data aggregator and ECG aggregator.

(a)   (b)
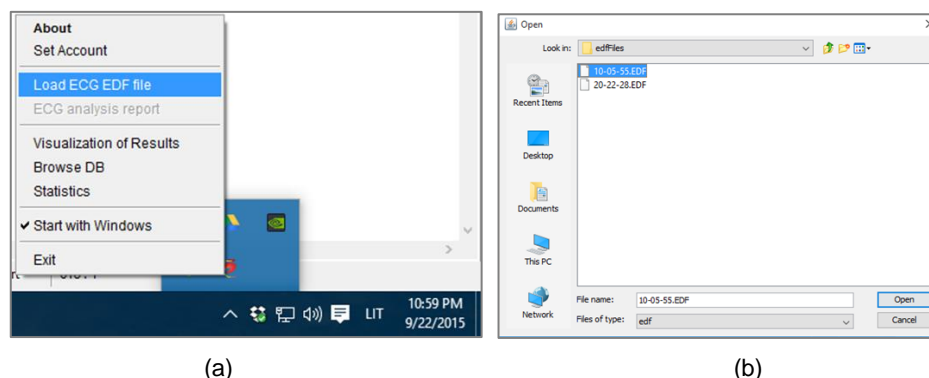
Figure 28. ECG EDF files are loaded from a tray menu of CARRE Lifestyle & ECG aggregator

Figure 29 shows the example of the ECG report generated after ECG file analysis. A patient can print his/her ECG report and show it to the physician during a health check visit.
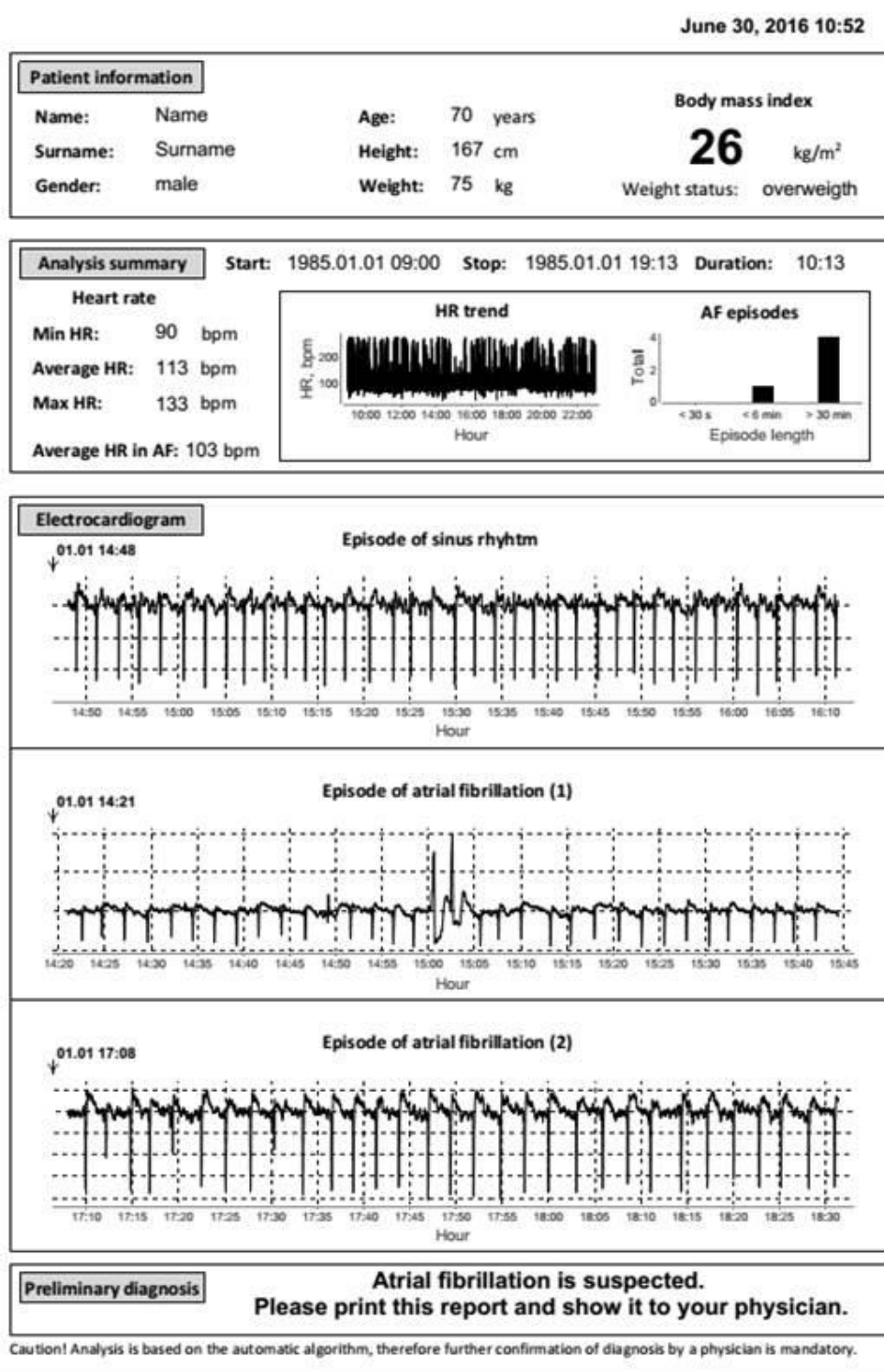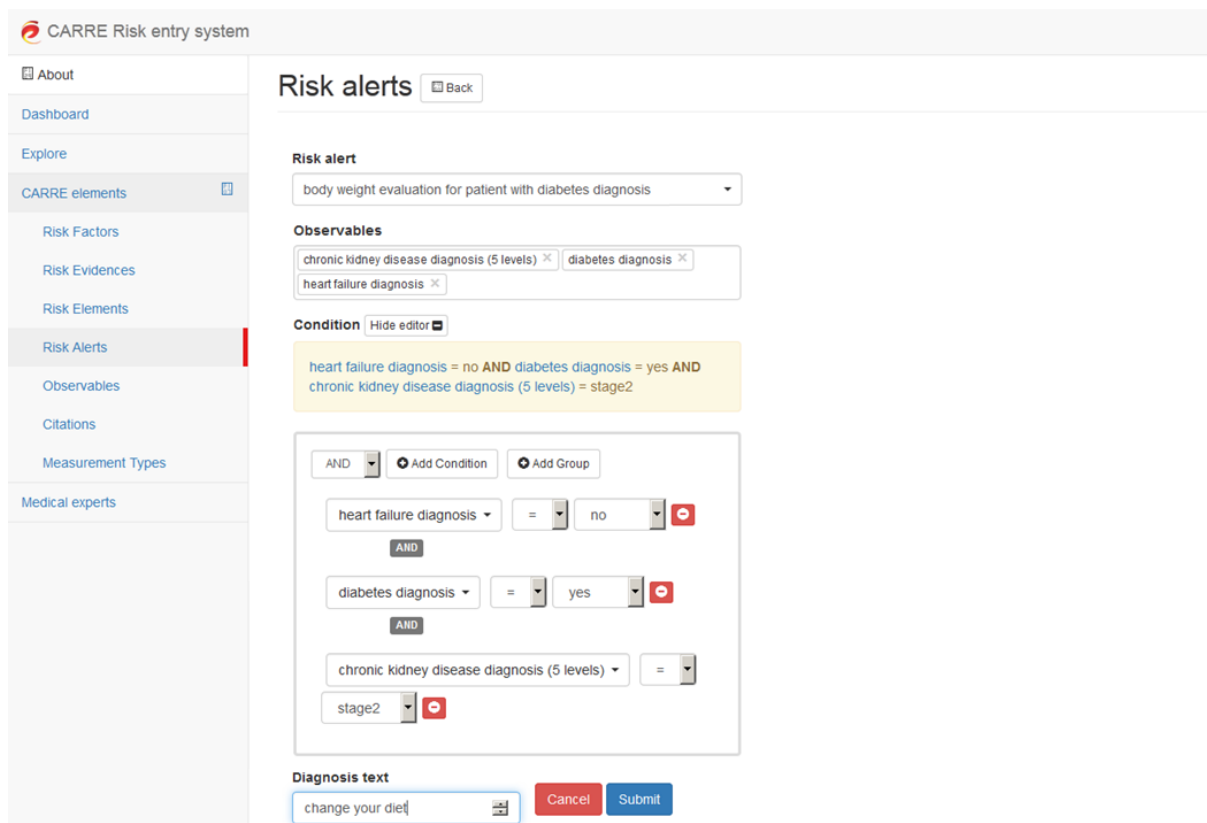
Figure 29. Report generated after ECG file analysis

5.5.1.6. Risk alert editing

A basic feature of the DSS alert system is to timely identificate the major concerns about patients' health condition levels. The DSS will send alerts, including medical check-ups, monitoring, increased risk of disease progression and transition, and suggestions to change diet.

The CARRE Decision Support is based on specifically designed algorithms that determine whether an alert will be executed for a particular patient condition. This is a result of analysis and reasoning captured in CARRE ontologies (CARRE risk factor database and/or the aggregated educational material). The DSS alert system outcome provides the variables to the CARRE visual interface for risk-alert visualisation; it is also possible to send e-mail alerts to the patient. Figure 30 shows the particular DSS alert editor.

Figure 30. DSS Alert Editor

5.5.1.7. Risk alert visualisation

The risk alerts calculated by the DSS module are stored in the private repository and can be accessed for risk alert visualisation by components in the visual interface. Figure 31 shows the process of risk alert visualisation after risk alert calculation.
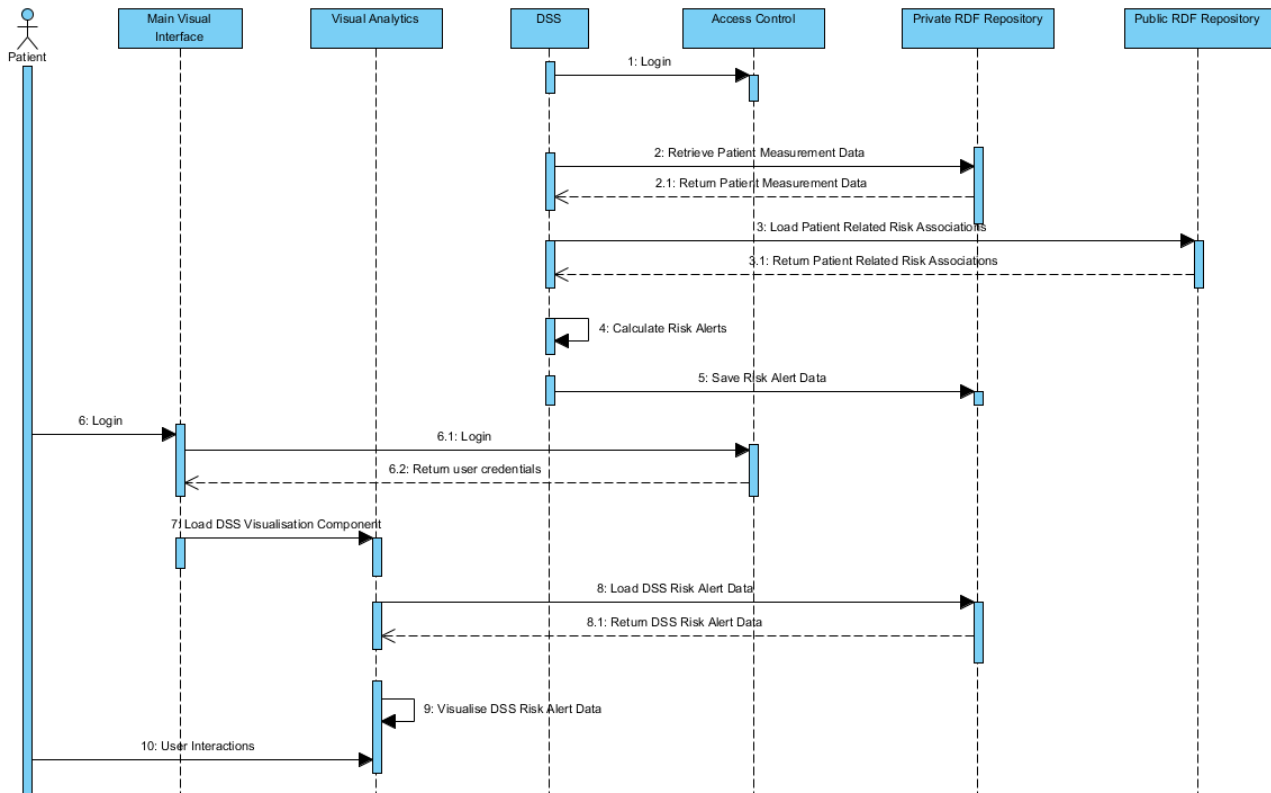
Figure 31. Sequence diagram of risk alert calculation and visualisation

5.5.2. Integrated user interface for medical professionals

5.5.2.1. The main visual interface

The main user interface for medical professionals shares many features with the patient application. One important difference is the Doctor menu under which tools and functions for medical professionals are listed, as shown in Figure 32, including patient management, risk entry, educational resource entry and the medical literature aggregator.
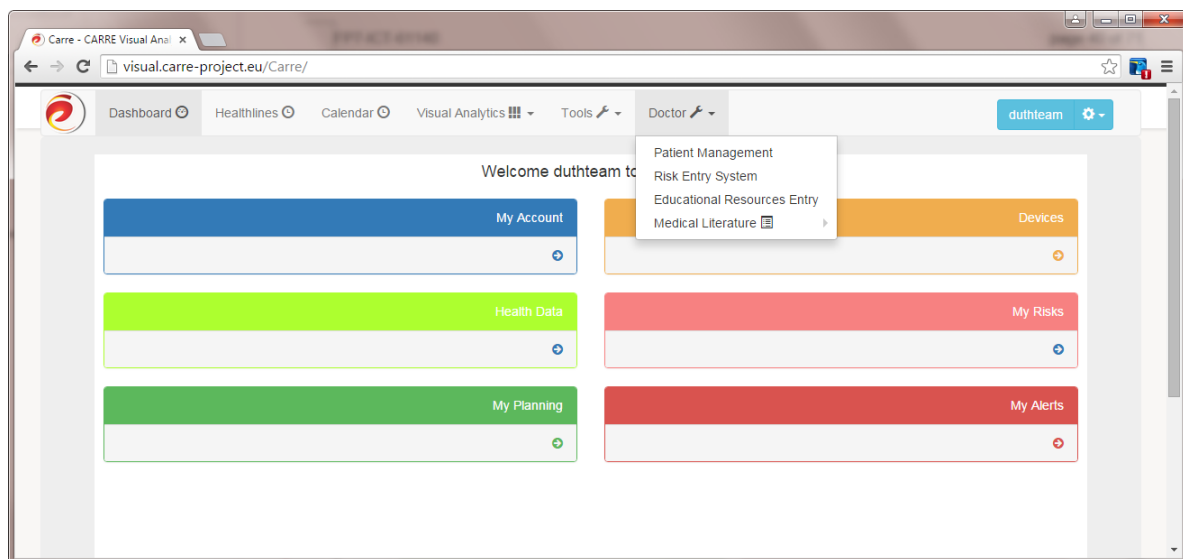
Figure 32. The Doctor menu for medical professionals

5.5.2.2. *Medical evidence aggregator visual interface*

The medical evidence aggregator gathers medical knowledge from medical publications to enrich the evidence of the existing risk descriptions and to identify new risk associations for cardiorenal diseases and comorbidity. The aggregator extracts and summarises key information from popular and trusted medical publications as they are indexed in PubMED. The medical evidence aggregator can be accessed from the main visual interface as shown in Figure 33. Figure 34 shows the user interface for the keyword based medical literature search. Figure 35 shows the user interface for keyword-based medical literature analysis.
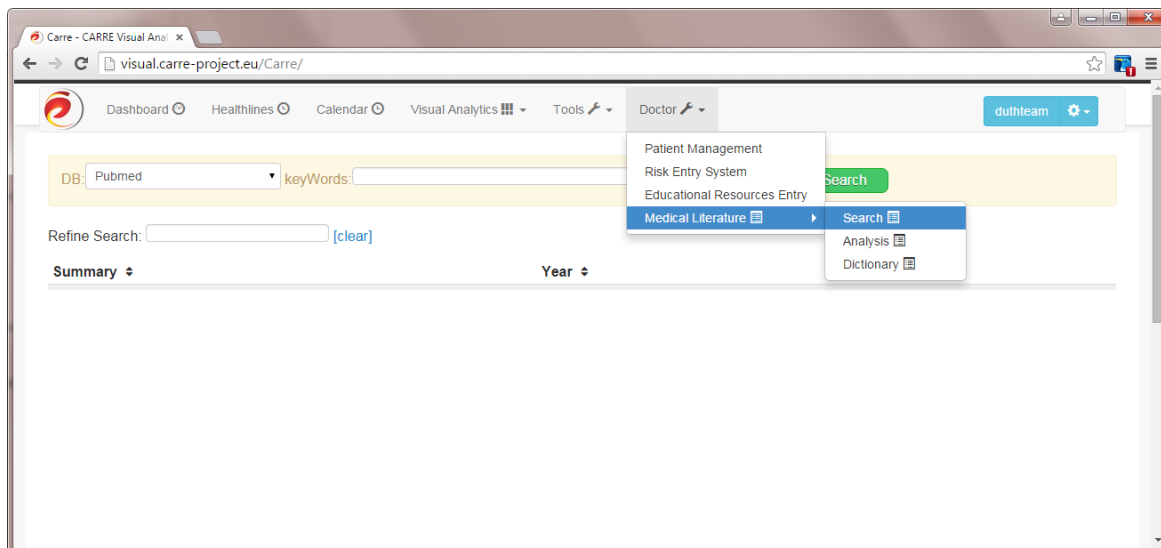
Figure 33. Accessing the medical evidence aggregator from the main visual interface
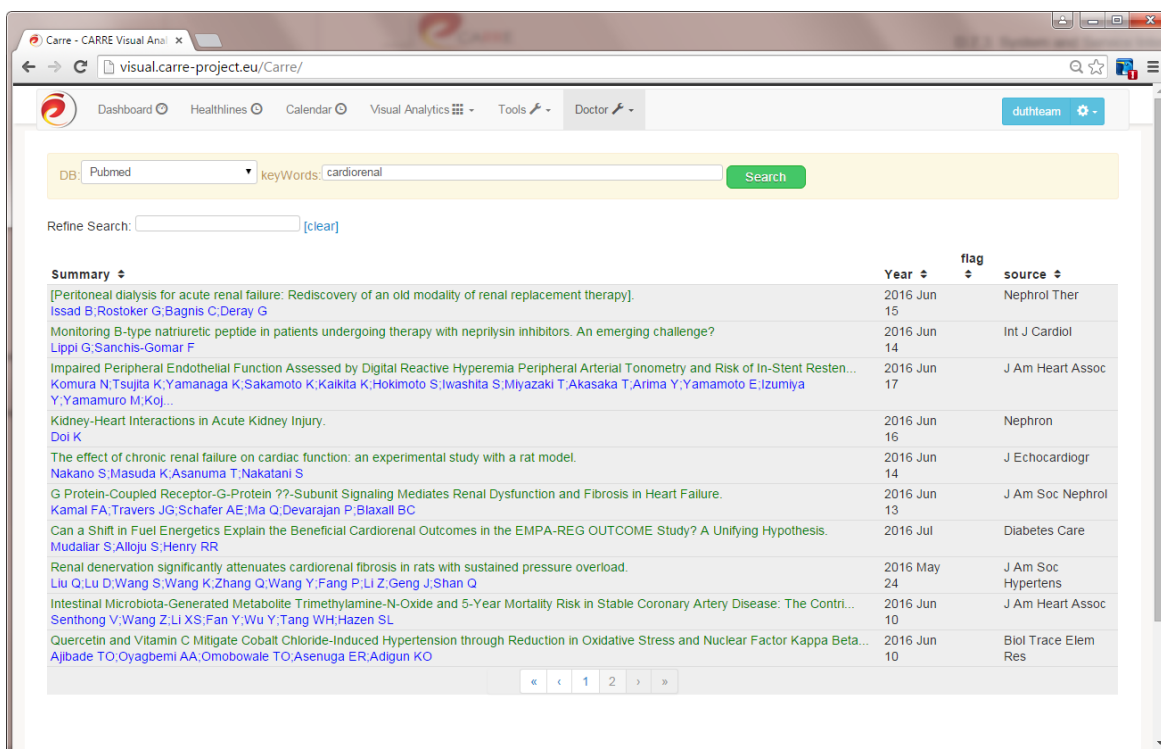
Figure 34. Medical literature search
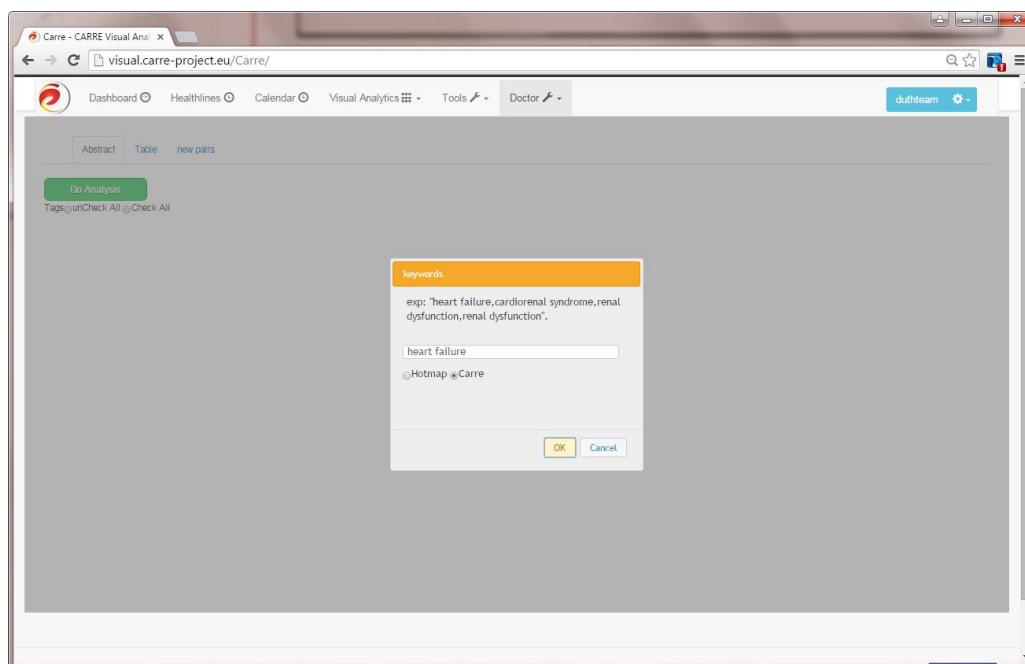
Figure 35. Literature analysis user interface

5.5.2.3. Risk factor entry

The risk association data extracted from the medical literature are stored in the CARRE public RDF repository. This is achieved by the doctors via the risk entry system, as shown in Figure 36. Figure 37 and Figure 38 are the user interfaces for inserting risk factors and risk evidence into the public RDF repository.
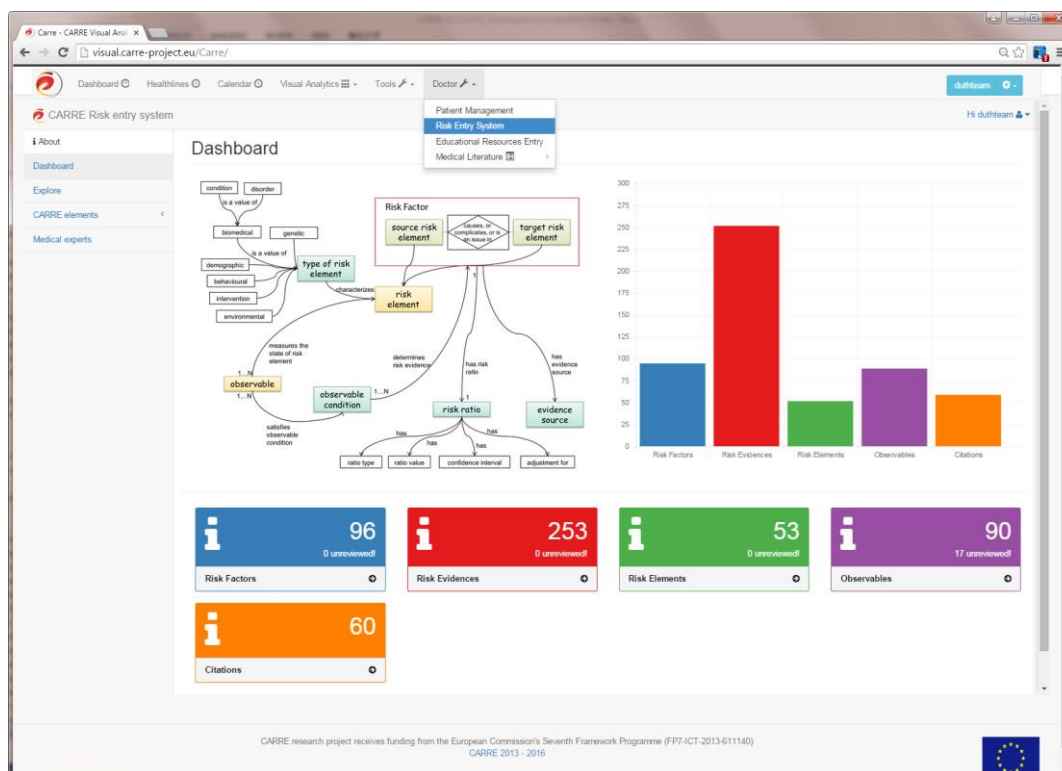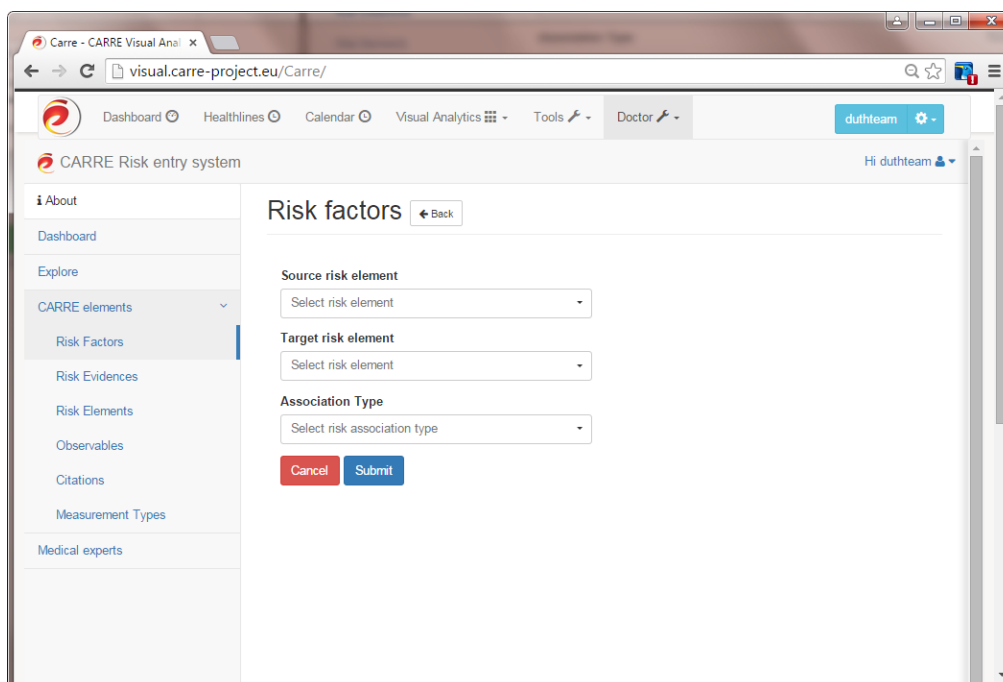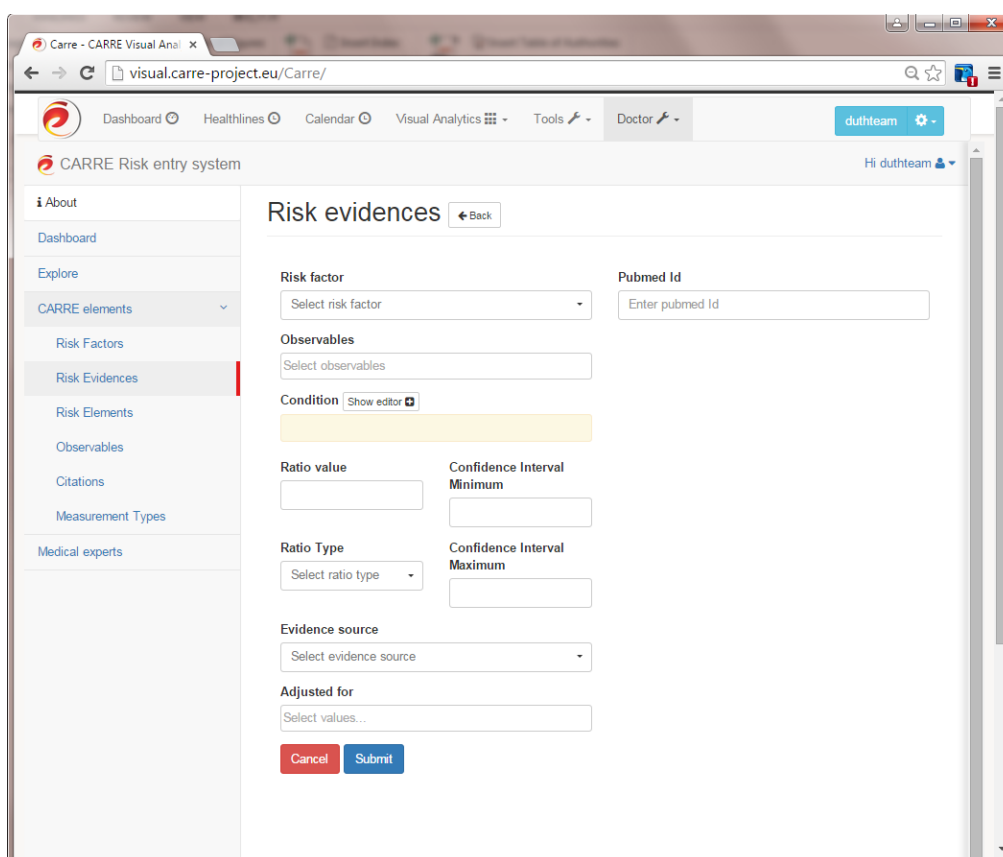
Figure 36. Risk entry system integrated in the main visual interface

Figure 37. Inserting risk factors

Figure 38. Inserting risk evidence

5.5.2.4. Risk factor visualization

The risk entry web interface provides not only the risk factor entry function but also the risk factor exploration and visualization to the medical professionals. Figure 39 shows the risk factor exploration user interface in the

risk entry system. Figure 40 shows visualisation of the risk factors in Sankey diagram for the doctors in the risk entry system.
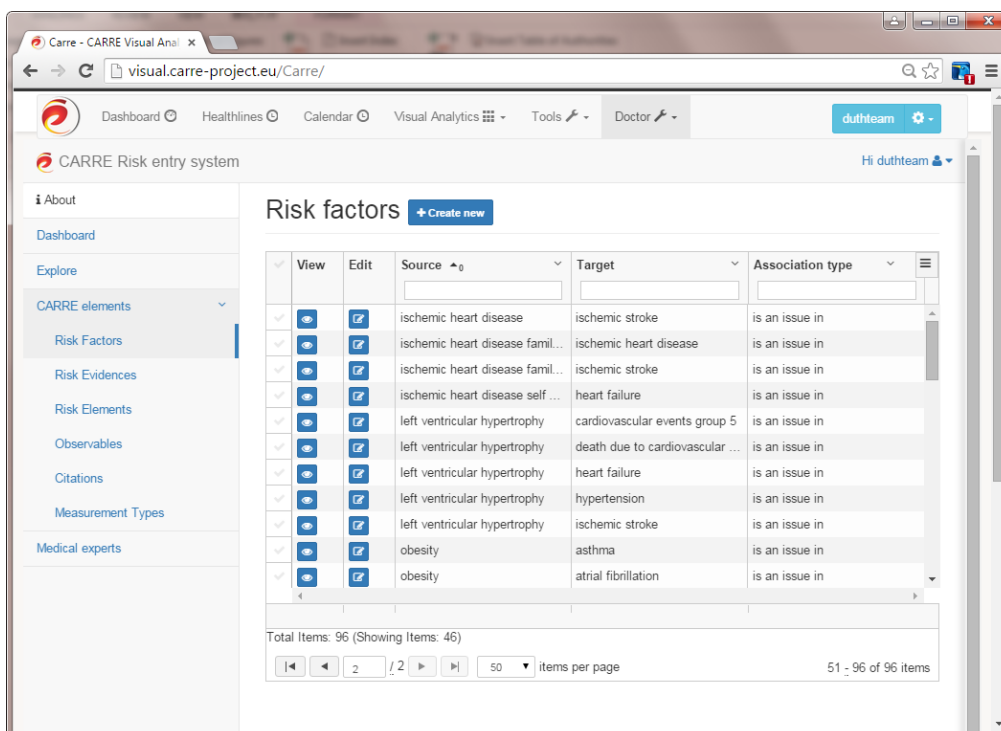
Figure 39. The risk factor exploration user interface
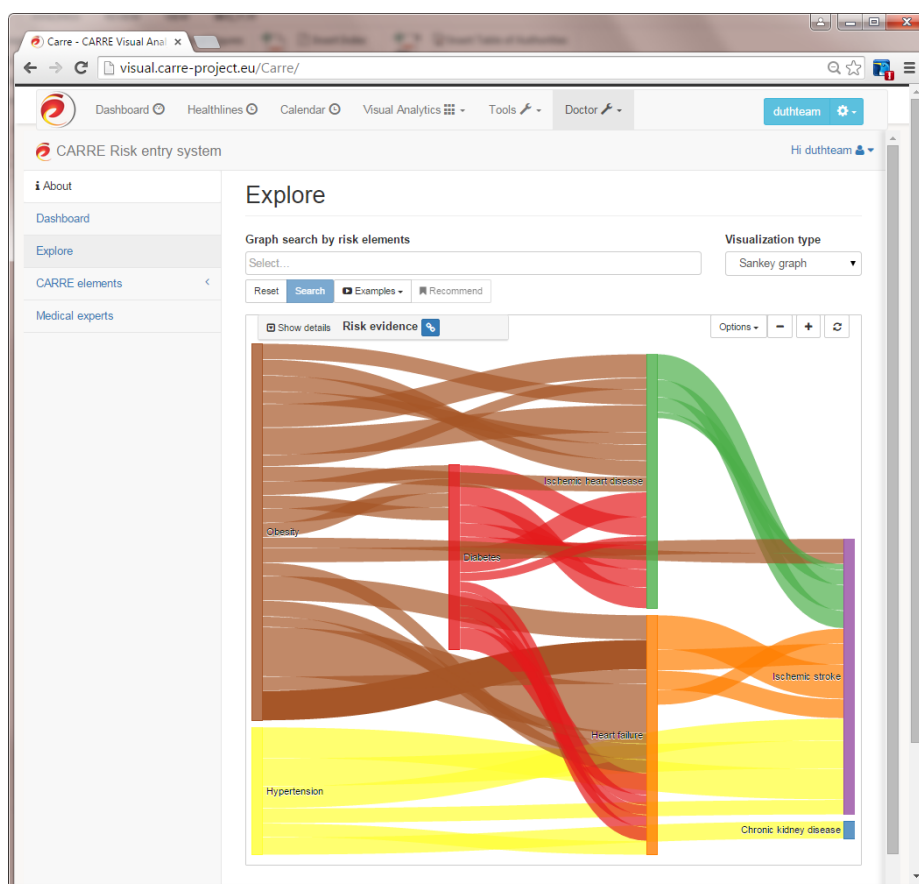
Figure 40. Visualisation of risk factors in Sankey diagram

6. System localisation

To address a wider user population, we need to support multiple languages. This section presents the approach used; clearly, it could be extended to support other languages.

6.1. Introduction

As an EU-based project, CARRE has partners and users across Europe. For evaluation, we also provided Greek and Lithuanian user interfaces to patients and doctors as CARRE has partners in the UK, Greece and Lithuania.

The localisation applies to the visual interface and to the data repositories. The visual interface localisation involves text messages in the HTML web pages while the data repository localisation involves key messages in the public and private data repositories, such as the name of the risk elements, etc.

This chapter describes the whole localisation procedure, including what we translate, how we prepare the translation, where we store the translated data, and how the live online translation has been achieved.

6.2. Key term localisation preparation

While online translation tools such as Google Translate provide automatic translation of the web pages and are easy to integrate into a website, the translations are often misleading or erroneous. Since it is critical to have accurate localisation of the terms used in the web interface for doctors and patients, manual translation of the terms is indispensable.

The procedure for manual translation is as follows:

1. Translation by first expert.
2. Independent translation by second expert.
3. Merging and final translation by third expert.
4. Backwards translation (to English) by fourth independent expert, without knowledge of the initial English text.
5. Merging of comments and final translation by another expert, who could be the same as Expert 3.

6.3. Key-term localisation

6.3.1. Introduction

The key-term localisation is based on the jQuery.i18n[9] javascript internationalisation/localisation library which simplifies the localisation of the web applications. After the key terms have been translated by domain experts, they are stored into jQuery.i18n message localisation JSON files, ready to be used by the CARRE components.

6.3.2. Message localisation JSON file

The jquery.i18n library uses a json based localisation file format, "banana", which is the localisation file format used by MediaWiki and other projects.

The translations of the key terms in different languages can be stored in a single file or in separate files.

A simple en.json file example from the jQuery.i18n website for the English language translation is given below:

```
{
```

---

[9]   https://github.com/wikimedia/jquery.i18n

```
            "@metadata": {
                "authors": [
                    "Alice",
                    "David",
                    "Santhosh"
                ],
                "last-updated": "2012-09-21",
                "locale": "en",
                "message-documentation": "qqq",
                "AnotherMetadata": "AnotherMedatadataValue"
            },
            "appname-title": "Example Application",
            "appname-sub-title": "An example application with jquery.i18n",
            "appname-header-introduction": "Introduction",
            "appname-about": "About this application",
            "appname-footer": "Footer text"
        }
```

It is also possible to put all the translation in a single file, as shown in the example below from the jQuery.i18n website. It is useful when the application is not large.

```
        {
            "@metadata": {
                "authors": [
                    "Alice",
                    "David",
                    "Santhosh"
                ],
                "last-updated": "2012-09-21",
                "locale": "en",
                "message-documentation": "qqq",
                "AnotherMetadata": "AnotherMedatadataValue"
            },
            "en": {
                "appname-title": "Example Application",
                "appname-sub-title": "An example application with jquery.i18n",
                "appname-header-introduction": "Introduction",
                "appname-about": "About this application",
                "appname-footer": "Footer text"
                },
            "ml": {
                "appname-title": "□□□□□□□□□□□□ □□□□□□□",
                "appname-sub-title": "jquery.i18n □□□□□□□□□□□□
    □□□□□□□□□□□□ □□□□□□□",
                "appname-header-introduction": "□□□□□",
                "appname-about": "ഇൗ□□□□□□□□□□□□□□□□□□□□□",
                "appname-footer": "□□□□□□□□□□□□□□□"
            }
        }
```

6.3.3. Usage

The following demonstration codes show the usage of jQuery.i18n library.

6.3.3.1. Setting the locale

```
    $.i18n( {
        locale: 'el' // Locale is Greece
    } );
```

6.3.3.2. Loading messages and message files

```
$.i18n().load( {
    en: {
        message_hello: 'Hello World',
        message_welcome: 'Welcome'
    },
    hi: 'i18n/messages_el.json', // Messages for Greek
    de: 'i18n/messages_lt.json'  // Messages for Lithuanian
} );
```

6.3.3.3. Message translation

The message translation can be achieved in several ways.

The translated text can be directly retrieved in javascript by the following method:

```
$.i18n( message_key );
```

It is also possible to use the data-i18n key to translate text embedded in HMTL tags without javascript:

```
<TAG data-i18n="message-key"></TAG>
```

6.3.4. Localisation JSON files for the CARRE project

This section presents the json localisation files used by the visual interface and visual analytics component.

6.3.4.1. Localisation of sample visual interface terms in English (va_en.json)

```
{
    "@metadata": {
        "authors": [
            "Allan Third"
        ],
        "last-updated": "2016-06-29",
        "locale": "en"
    },
    "MENU_dashboard": "Dashboard",
    "MENU_health_lines": "Healthlines",
    "MENU_scheduler": "Calendar",
    "MENU_disease_progression": "Progression of disease",
    "MENU_components": "Components",
    "MENU_login": "Login",
    "MENU_educational_resources": "Educational Resources Entry",
    "DASHBOARD_my_planning": "My planning",
    "DASHBOARD_my_alerts": "My alerts",
    "DASHBOARD_welcome_message": "Welcome to CARRE Visual Analytics",
    "DASHBOARD_my_account": "My account",
    "DASHBOARD_devices": "Devices",
    "DASHBOARD_health_data": "Health data",
    "DASHBOARD_my_risks": "My risks",
    "SCHEDULER_year": "Year",
    "SCHEDULER_month": "Month",
    "SCHEDULER_title": "CARRE kalendorius",
    "COMPONENTS_data_correlation": "Data Correlation",
    "COMPONENTS_matrix_view": "Matrix",
    "COMPONENTS_chord_view": "Chord Diagram",
    "COMPONENTS_sankey_view": "Sankey Diagram",
    "COMPONENTS_graph_view": "Graph",
    "COMPONENTS_risk_factor_matrix_view": "Matrix",
    "HEALTHLINE_physical_activity": "Physical Activity",
    "HEALTHLINE_blood_pressure": "Blood Pressure",
    "HEALTHLINE_blood_sugar": "Blood Sugar",
```

```
        "HEALTHLINE_weight": "Weight",
        "MENU_va": "Visual Analytics",
        "MENU_Patient_Risk_Graph": "Patient Risk Graph",
        "MENU_doctors": "Doctor",
        "MENU_profile": "Profile",
        "MENU_search": "Search",
        "MENU_analysis": "Analysis",
        "MENU_dictionary": "Dictionary",
        "MENU_medical_literature": "Medical Literature",
        "MENU_tools": "Tools",
        "MENU_device": "Devices",
        "MENU_risk_entry_system": "Risk Entry System",
        "MENU_patient_management": "Patient Management",
        "MENU_phr_entry": "PHR Entry",
        "MENU_help": "Help",
}
```

6.3.4.2. Localisation of sample visual interface terms in Greek (va_el.json)

```
{
        "@metadata": {
           "authors": [
                "Nick Portokallidis",
                "Tina Pouliliou",
                "George Drosatos",
                "Spyros Kavvadias",
                "Eleni Kaldoudi"
           ],
           "last-updated": "2016-06-29",
           "locale": "el"
        },
        "MENU_dashboard": "Αρχική Σελίδα",
        "MENU_health_lines": "Γραμμές Υγείας",
        "MENU_scheduler": "Ημερολόγιο",
        "MENU_disease_progression": "Παράγοντες Κινδύνου",
        "MENU_components": "Διαγράμματα",
        "MENU_login": "Σύνδεση",
        "MENU_profile": "προφίλ",
        "DASHBOARD_my_planning": "Προγραμματισμός",
        "DASHBOARD_my_alerts": "Οι ειδοποιήσεις μου",
        "DASHBOARD_welcome_message": "Καλώς ήρθατε στο CARRE",
        "DASHBOARD_my_account": "Ο λογαριασμός μου",
        "DASHBOARD_devices": "Συσκευές",
        "DASHBOARD_health_data": "Δεδομένα της υγείας μου",
        "DASHBOARD_my_risks": "Παράγοντες κινδύνου",
        "SCHEDULER_year": "Έτος",
        "SCHEDULER_month": "Μήνας",
        "SCHEDULER_title": "Ημερολόγιο CARRE",
        "COMPONENTS_data_correlation": "Συσχέτιση παραγόντων κινδύνου",
        "COMPONENTS_matrix_view": "Πίνακας",
        "COMPONENTS_chord_view": "Διάγραμμα τόξου",
        "COMPONENTS_sankey_view": "Διάγραμμα ροής",
        "COMPONENTS_graph_view": "Γράφος",
        "COMPONENTS_risk_factor_matrix_view": "Πίνακας παραγόντων κινδύνου",
        "HEALTHLINE_physical_activity": "Φυσική δραστηριότητα",
        "HEALTHLINE_blood_pressure": "Πίεση αίματος",
        "HEALTHLINE_blood_sugar": "Επίπεδα σακχάρου στο αίμα",
        "HEALTHLINE_weight": "Βάρος",
        "MENU_va": "Visual Analytics",
        "MENU_Patient_Risk_Graph": "Γράφημα κινδύνου του ασθενούς",
        "MENU_doctors": "Γιατρός",
```

```
        "MENU_profile": "Προφίλ",
        "MENU_search": "Έρευνα",
        "MENU_analysis": "Ανάλυση",
        "MENU_dictionary": "Λεξικό",
        "MENU_medical_literature": "Ιατρική βιβλιογραφία",
        "MENU_tools": "Εργαλεία",
        "MENU_device": "Συσκευές",
        "MENU_risk_entry_system": "Σύστημα εισαγωγής παραγόντων κινδύνου",
        "MENU_patient_management": "Διαχείριση ασθενών",
        "MENU_phr_entry": "Διαχείριση φακέλου υγείας ασθενούς",
        "MENU_help": "Βοήθεια"
    }
```

6.3.4.3. Localisation of sample visual interface terms in Lithuanian (va_lt.json)

```
    {
        "@metadata": {
            "authors": [
                "Laurynas",
                "Gintarė",
                "Neringa",
                "Domantas"
            ],
            "last-updated": "2016-06-29",
            "locale": "lt"
        },
        "MENU_dashboard": "Įrankiai",
        "MENU_health_lines": "Sveikatos parametrų kreivės",
        "MENU_scheduler": "Kalendorius",
        "MENU_disease_progression": "Ligos progresavimas",
        "MENU_components": "Komponentai",
        "MENU_login": "Prisijungimas",
        "MENU_profile": "Profilis",
        "DASHBOARD_my_planning": "Mano planai",
        "DASHBOARD_my_alerts": "Mano pokyčiai",
        "DASHBOARD_welcome_message": "Sveiki atvykę į Carre vaizdinę analizę",
        "DASHBOARD_my_account": "Mano paskyra",
        "DASHBOARD_devices": "Prietaisai",
        "DASHBOARD_health_data": "Informacija apie sveikatą",
        "DASHBOARD_my_risks": "Mano rizika",
        "SCHEDULER_year": "Metai",
        "SCHEDULER_month": "Mėnuo",
        "SCHEDULER_title": "CARRE kalendorius",
        "COMPONENTS_data_correlation": "Duomenų koreliacija",
        "COMPONENTS_matrix_view": "Matricos vaizdas",
        "COMPONENTS_chord_view": "Juostinis vaizdavimas",
        "COMPONENTS_sankey_view": "Sankey vaizdas",
        "COMPONENTS_graph_view": "Diagramos",
        "COMPONENTS_risk_factor_matrix_view":   "CARRE    rizikos    veiksnių
matrica",
        "HEALTHLINE_physical_activity": "Fizinė veikla",
        "HEALTHLINE_blood_pressure": "Kraujospūdis",
        "HEALTHLINE_blood_sugar": "Gliukozės kiekis kraujyje",
        "HEALTHLINE_weight": "Svoris",
        "MENU_va": "Vaizdinė Analizė",
        "MENU_Patient_Risk_Graph": "Paciento rizikos diagrama",
        "MENU_doctors": "Gydytojas",
        "MENU_profile": "Profilis",
        "MENU_search": "Paieška",
        "MENU_analysis": "Analizė",
        "MENU_dictionary": "Žodynas",
```

```
            "MENU_medical_literature": "Medicininė literatūra",
            "MENU_tools": "Įrankiai",
            "MENU_device": "Prietaisai",
            "MENU_risk_entry_system": "Rizikos įrašas",
            "MENU_patient_management": "Paciento gydymas",
            "MENU_phr_entry": "Paciento sveikatos įrašas",
            "MENU_help": "Padėti"
        }
```

6.4. Localisation in the visual interface

6.4.1. Initialisation

In the visual interface main HTML page template/layout.html, the following javascript code sets the locale and loads the the key term translations.

```
        $.i18n({ locale: carre_language });
        $.i18n().load( {
            el: 'static/share/i18n/va_en.json',
            el: 'static/share/i18n/va_el.json',
            lt: 'static/share/i18n/va_lt.json'
        } ).done(function() {
            $("body").i18n();
        });
```

6.4.2. Key term translation

The translation of the key terms in the visual interface includes static term translation and dynamic term translation. The static terms do not change during the lifecycle of the web-based visual interface while the dynamic terms are dynamically generated by the javascript code running in the web-based visual interface. The static and dynamic term translation are both supported by jQuery.i18n through different methods which are introduced as follows.

6.4.2.1. Static term translation

In the main visual interface HTML page template/layout.html, most of the menu items represent static links to visual analytics components, medical evidence functions and other CARRE components. They are static messages that can be translated by the jQuery.i18n using the following data-i18n method.

```
        <TAG data-i18n="MESSAGE"> </TAG>
```

The TAG represents an HTML tag which contains text, for example

```
        <span data-i18n="MENU_dashboard"> </span>
```

or

```
        <li data-i18n="MENU_search"> </li>
```

One can use default messages in the content so that one still sees the default visual interface even the jQuery.i18n is not working, for example:

```
        <span data-i18n="MENU_calendar">Calendar</span> ,
```

or one can just leave them blank for jQuery.i18n to fill in the translated messages at run time.

For static key terms in the web-based interface of other components, such as those in the web interface of the visual analytics components or risk entry system, the localisation can be achieved in a similar way.

6.4.2.2. Dynamic term translation

For dynamic web pages, the contents of a web page are often generated dynamically so that one cannot know the message keys for jQuery.i18n in advance. Under such cases, the data-i18n method presented in the previous section cannot be used, and the dynamic method in javascript and jQuery should be used in the following way:

```
$.i18n( message_key );
```

The message_key is the dynamically generated message key for jQuery.i18n. For example, the text dynamically showing on the login button can either be "Login" or the user's name, the localisation of the "Login" text message is done via

```
$.i18n("Login");
```

Figure 41 shows the main visual interface in Greek.
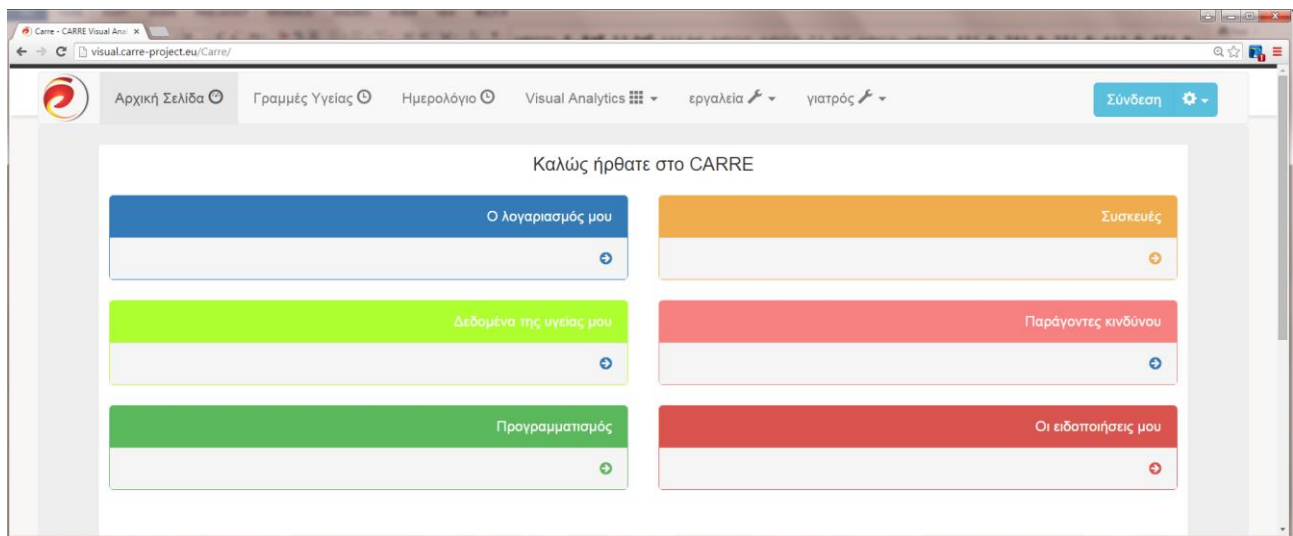
Figure 41. The main visual interface and dashboard in Greek

6.5. Localisation in the RDF repository

In the RDF repository the following concepts have been enriched with Greek and Lithuanian language tags (el,lt) :

- Risk Element (#risk_element) , translated properties:
    - #has_risk_element_name
- Observable (#observable) , translated properties:
    - #has_observable_name
- Measurement type (#measurement_type) , translated properties:
    - #has_measurement_type_name
    - #has_label
    - #has_enumeration_values

The detailed RDF translations are listed in Annex A : RDF Translations.

Example query for loading the Greek translation of the risk element names from Public RDF

```
SELECT * FROM <http://carre.kmi.open.ac.uk/riskdata> WHERE {
?element a risk:risk_element.
?element risk:has_risk_element_name ?rl_name.

    FILTER(langMatches(lang(?rl_name), 'el' ) )

}
```

6.6. Localisation in visual analytics

The localisation in visual analytics is based on the key-term translation and the RDF repository localisation.

6.6.1. General information localisation

The general information in the HTML web pages or the javascript file is done with the jQuery.i18n library as introduced in Section 6.3.

6.6.2. Measurement variable localisation

In healthline visualisation, the measurement names need to be localised. The private RDF repository supports localisation of the measurement names. The visual analytics component can query the localised measurement variable names by filtering the specified language in the SPARQL query in the following way:

```
SELECT DISTINCT ?measurementtype ?ob_name
FROM <https://carre.kmi.open.ac.uk/users/user.name>
FROM <http://carre.kmi.open.ac.uk/riskdata> WHERE
    ?measurement ?measurementtype ?object .
:has_blood_pressure_systolic :has_weight :has_has_blood_glucose_fasting}
    OPTIONAL {?ob a risk:observable; risk:has_external_predicate
?measurementtype; risk:has_observable_name ?ob_name.  }
    FILTER(langMatches(lang(?ob_name), carre_language ) )
```

where carre_language is the human language used by the system.

6.6.3. Risk element and observable name localisation

In risk graph visualisation, the risk element and risk observable names need to be localised. The public RDF repository supports localisation of the measurement names. The query of risk element list and risk observable list is performed via the RESTful APIs which provide a language parameter to select the desired language for the element or observable names. The risk element query URL is as follows:

```
var url =
"https://carre.kmi.open.ac.uk/ws/instances?type=risk_element&language=" +
carre_lanaguage;
```

The returns of the query includes triples of names in the des

```
object: "ūminis inkstų pažeidimas"
predicate:
"http://carre.kmi.open.ac.uk/ontology/risk.owl#has_risk_element_name"
subject: http://carre.kmi.open.ac.uk/risk_elements/RL_1
```

The query of risk observable names in the local language can be achieved in a similar way.

Figure 42 shows the patient risk graph with risk elements in Lithuanian.

Figure 42. The patient risk graph in Lithuanian

6.6.4. Date localisation

Visual Analytics uses d3.js to generate interactive visualisations in the web page, however, the support for localisation of d3.js is not perfect. For example in the healthline visualisation, there is no native support from the d3.js library for localisation of dates. In this situation, date localisation is achieved by manually defining the d3.js customised date-format support in the following way:

```
d3Locales = {
"lt" :
{
    "days": [ "Sekmadienis", "Pirmadienis", "Antradienis", "Trečiadienis",
"Ketvirtadienis", "Penktadienis", "Šeštadienis" ],
    "shortDays": ["Se", "P", "A", "T", "K", "Pn", "Še"],
    "months": [ "Sausis", "Vasaris", "Kovas", "Balandis", "Gegužė",
"Birželis", "Liepa", "Rugpjūtis", "Rugsėjis", "Spalis", "Lapkritis",
"Gruodis" ],
    "shortMonths": ["01", "02", "03", "04", "05", "06", "07", "08", "09",
"10", "11", "12" ]
},

"el" :
{
    "days": [ "Κυριακή",  "Δευτέρα", "Τρίτη", "Τετάρτη", "Πέμπτη",
"Παρασκευή", "Σάββατο"],
    "shortDays": [ "Κυρ", "Δευ", "Τρι", "Τετ", "Πεμ", "Παρ", "Σαβ" ],
    "months": [ "Ιανουάριος", "Φεβρουάριος", "Μάρτιος", "Απρίλιος",
"Μάιος", "Ιούνιος", "Ιούλιος", "Αύγουστος", "Σεπτέμβριος", "Οκτώβριος",
"Νοέμβριος", "Δεκέμβριος" ],
    "shortMonths": [ "Ιαν", "Φεβ", "Μαρ", "Απρ", "Μάι", "Ιουν", "Ιουλ",
"Αυγ", "Σεπ", "Οκτ", "Νοε", "Δεκ" ]
```

```
},
"en" :
{
    "days": ["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday",
"Friday", "Saturday"],
    "shortDays": ["SUN", "MON", "TUE", "WED", "THU", "FRI", "SAT"],
    "months": ["January", "February", "March", "April", "May", "June",
"July", "August", "September", "October", "November", "December"],
    "shortMonths": ["Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul",
"Aug", "Sep", "Oct", "Nov", "Dec"]
}}
```

And set the local for d3 date :

```
var locale = d3Locales[carre_language];
```

As moment.js provides global locale setting and localisation control, in the calendar we use moment.js for localisation of the week and month names in the following way.

```
moment.locale(carre_language);
for (var i = 1; i <= 7; i++)
    localisedWeekDayNames.push(moment.weekdaysShort(i));
monthNames = moment.months();
```

Figure 43 shows an example of Healthlines in Greek – the variable names and the month names in the timelines are in Greek.
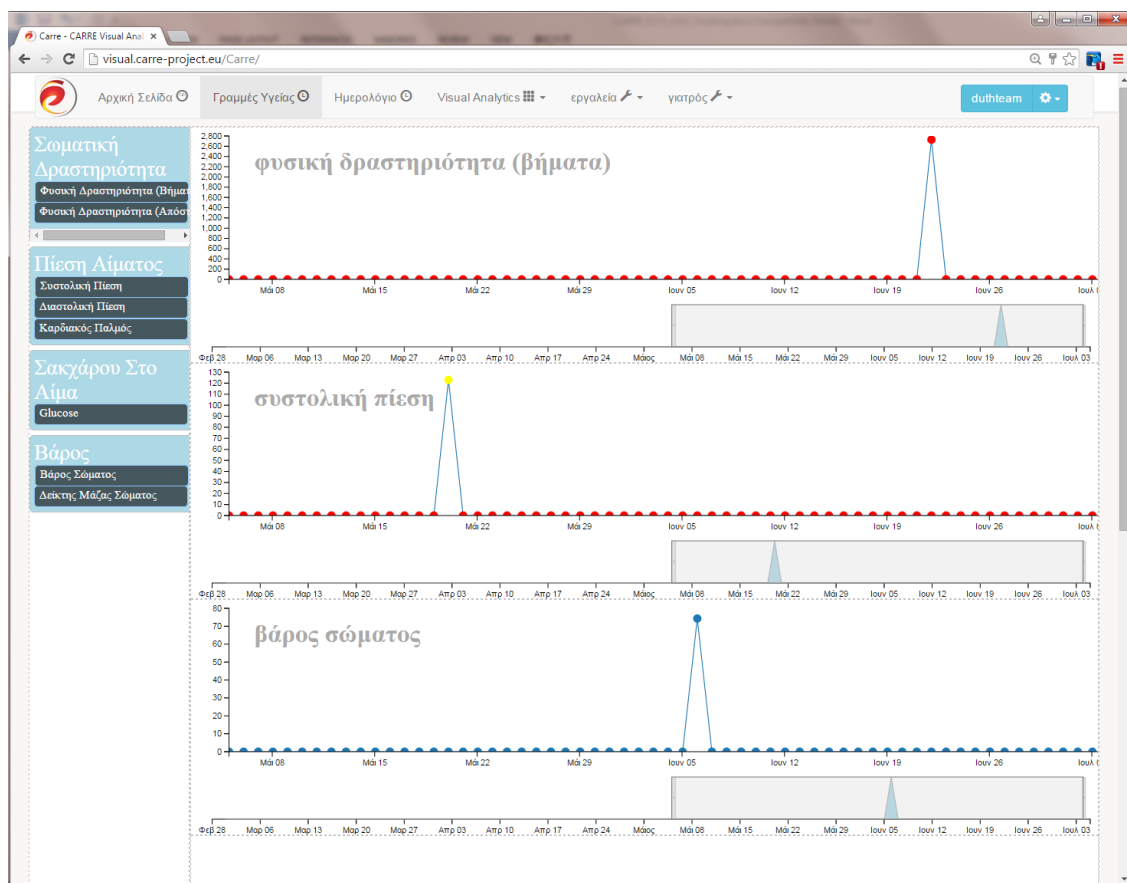
Figure 43. Healthlines in Greek

7. System deployment

This section provides information on the deployment of each CARRE subsystem.

7.1. Visual interface and medical evidence aggregator server

The visual interface and medical evidence systems are two relatively independent subsystems which can be deployed separately. The visual interface system is used by doctors and patients while the medical evidence system is used mainly by doctors. At this stage, they are in the same web container for ease of deployment and management as well as for saving computer resources.

7.1.1. Server requirements

The hardware and software requirements for the server of visual interface and medical evidence aggregators are listed in the following table:

Table 2. Server requirements for the visual interface and medical evidence aggregator.	
Hardware requirements	
CPU	6 Cores, > 2.8 GHz
Memory	8GB
DB	RDF repository Virtuoso
Hard Drive	192GB SSD storage
Software requirements	
Operating System	Linux, Ubuntu 14.04 LTS
Environment	Apache Tomcat 7 server Java 8 spring framework 4.1 Gate 8 and plugins clearNLP 2.0

7.1.2. General steps of the deployment process

As the visual interface and the medical evidence aggregator share the same web server, their deployment is introduced in one section.

7.1.2.1. Environment setup

1. Make sure you have Java 8 installed.

```
java -version
```

2. Install a git client, (as we use git to do source code control, make sure you have the administrator privilege):

```
apt-get install git
```

3. Install Maven 3.x.x for ubuntu, which is our building tool:

```
apt-get install maven
```

4. Install sesame RDF, by just copying openrdf-sesame and workbench jars under Tomcat webapps directory.

7.1.2.2. Initial build and run

1. Under the command line, go to a selected working folder, and get the latest source:

```
git clone https://weihuiBeds@bitbucket.org/weihuiBeds/doc_repos_ccgv.git
```

2. Your git url may be different, please login bitbucket to find the url of your own project, then check out branch "carre".

```
git checkout –b carre
```

3. Go to the source code folder.

```
cd doc_repos_ccgv
```

4. Run maven command to build the package file (this may take minutes on initial build)

```
mvn clean package
```

5. If no exception appears on the console, copy the .war package file to the tomcat directory

```
cp target/Carre.war /var/lib/tomcat7/webapps/
```

6. Remove the old files and restart tomcat

```
rm -rf /var/lib/tomcat7/webapps/Carre
/etc/init.d/tomcat7 restart
```

7. Then acess the following URL from the browser

```
http://visual.carre-project.eu/Carre/
```

8. You also need to load RDF data into system Sesame repository by using system functions under Manage -> load keywords.

7.1.2.3. Application update

After the application is installed there may be occasions to update the application due to function improvements or bug fixes. The process of application update is as follows which is generally repeating steps 3~6 in Section 7.1.2.2.

1. Login to the server via SSH
2. Update the source code of the visual interface and medical evidence aggregator from the bitbucket repository
3. Run maven command to re-generate the .war package file
    mvn clean package
4. Copy the .war package file to the apache tomcat directory
5. Remove the old files and restart tomcat

The deployment process on the server has been written into one script as the follows to ease deployment.

```
git pull
mvn clean package
sudo cp target/Carre.war /var/lib/tomcat7/webapps/
sudo rm -rf /var/lib/tomcat7/webapps/Carre
sudo /etc/init.d/tomcat7 restart
```

7.1.2.4. Deploy locally for debugging

Sometimes the system needs to be deployed locally for developing, debugging or other purposes. To make a local deployment, the environment setup is the same as that in section 7.1.2.1. The building and running process is slightly different from that in 7.1.2.2. The steps are listed as follows:

1. Enter the source code folder.

```
cd doc_repos_ccgv
```

2. Run maven command to build the package file (this may take minutes on initial build)

```
mvn clean tomcat7:run -DENV_SYSTEM=.dev
```

3. If no exception is thrown from the command, it is ready to visit the following URL:

```
http://127.0.0.1:8080/Carre
```

4. If the developer likes to use it with OU Authentication, they need to make a mapping in the host file /etc/hosts by adding the following line:

```
127.0.0.1 visual.carre-project.eu
```

5. Now it is ready to visit the system with the following URL in the browser

```
http://visual.carre-project.eu/Carre/
```

7.2. Authentication, data aggregation, decision support and storage server

The data aggregators (sensor, PHR and management of them), access control component (including RESTFul API), decision support service and RDF repositories (public and privte) are located in the same server. This happens for performance and management reasons as well as for saving computer resources.

7.2.1. Server requirements

The hardware and software requirements for the server of data aggregators, access control (RESTFul API), decision support service and RDF repositories are listed in the following table:

Table 3. Server requirements for the authentication, aggregation and data repository.	
Hardware requirements	
CPU	8 Cores, > 2.2 GHz
Memory	16GB
DB	RDF repository Virtuoso
Hard Drive	> 120GB storage
Software requirements	
Operating System	Linux, Redhat Enterprise 6.7
Environment	Apache Tomcat 6 server, Java 8, Python 2.7, Visrtuoso 6

7.2.2. General steps of the deployment process

This component will be provided as a mainly pre-configured virtual machine image. Deployment details will vary depending on the particular virtualisation environment being used. Once running as a virtual server, it will be possible to connect via SSH to upload a deployment configuration file (a template will be made available in the Breakthroughs section of the CARRE website, in order to inform the new deployment of its relevant hostname, desired administration authentication details, and so on).

The sensor data aggregators each require individual configuration with the relevant device manufacturers, following their own up-to-date instructions for creating and managing an API client application. This process will provide application keys and details which can be entered into the template configuration file to be uploaded to the deployed server. This is necessary for each separate deployment so that each manufacturer is provided with the relevant server URLs in order to send data to the correct destination for users of the deployment.

7.3. PHR manual entry system server

The PHR manual entry system was developed using .NET framework and for this reason it requires a server with Windows operating system.

7.3.1. Server requirements

The hardware and software requirements for the server of PHR manual entry system are listed in the following table:

Table 4. Server requirements for PHR entry system.	
Hardware requirements	
CPU	2 Cores, > 3.0 GHz
Memory	4GB (or more)
Hard Drive	250 GB (or more)
Software requirements	
Operating System	Windows Server 2012
Server	IIS 7.0
Database	Microsoft SQL server 2012
Other	NET framework v4.5.1, NuGet package manager, Visual Studio 2013

7.3.2. General steps of the deployment process

Visual Studio is used to facilitate the deployment as it provides abstractions for default Windows procedures.

7.3.2.1. Preparing for deployment

In the folder containing the PHR entry system project is a file named 'ManualDataEntry.sln' which contains solution information; it can be opened using Visual Studio 2013.

By clicking Build -> Build Solution, Visual Studio restores all dependency packages and compiles the project. Alternatively this can be done by:

```
# navigate to root of solution directory where ManualDataEntry.csproj file is
    located.
```

```
nuget restore
#this restores all the needed dependencies
msbuild ManualDataEntry.csproj
```

After a successful build, these commands should be run in the nuget package manager console:

```
# ensure that database has all the required tables (using code first migrations)
enable-migrations
update-database
# now all the required tables should be present in database
```

Now the project can be deployed using the settings found in Prod.pubxml file by clicking Build -> Publish ManualDataEntry -> Publish.

7.3.2.2. Deployment configuration

Deployment configuration is found in 2 files: *Web.config* and *Properties\PublishProfiles\Prod.pubxml*. Web.config is an XML file that deals with environment variables and database connections. The database connection can be altered by modifying connectionStrings->add element with attribute "name=DefaultConnection". In this file there can also be found these configurations inside of "appSettings" node: "loginUrl" – url where user is redirected if he is not logged into the CARRE project; "logoutUrl' – url where the user is redirected when he wishes to logout from the CARRE project. Prod.pubxml is an XML file that contains information about where the project should be deployed and what build configuration for deployment should be used. This file can be edited with a plain text editor, but there is an option to edit with graphical interface from Visual Studio by clicking Build->Publish ManualDataEntry.

7.4. CARRE Device system

7.4.1. Client requirements of the CARRE ECG aggregator

The hardware and software requirements for the client PC machine running a background service CARRE ECG (and Lifestyle) aggregator interface are listed in the Table 5. The long term ECG records (<72 h) obtained with the selected ECG devices (Emotion Faros 180; Mega Electronics Ltd, Finland) can be as large as 950 MB. Therefore, the CARRE ECG aggregator application (developed in JAVA) has main requirement – adequate amount of RAM.

Table 5. Client requirements for CARRE ECG aggregator

Hardware requirements	
RAM memory	8 GB or more
Software requirements	
Platform	Windows, Java 8 JRE
Browser	Google Chrome or Firefox Desktop
Input devices	Keyboard and mouse
Other	Internet Access

7.4.2. General steps of deployment process

CARRE multiparametric weight scales collect data and automatically transfer the records as an encrypted GDF+ files via WiFi connection to the remote server on the Internet. The present configuration and deployment of components is shown in Figure 44. The data record includes: force sensor data (for weight), 3 lead electrocardiogram, two electrically independent bioimpedance channels and environmental data. In order to perform visualisation and analysis of acquired data, a prototype software application was developed (Figure 45). The application is able to process acquired signals and to present extracted parameters such as:

− Weight;

− Body composition parameters (segmental bioimpedance magnitude and phase angle for the range of frequencies);

− Heart rate trend and heart arrhythmia (atrial fibrillation) detection results;

− Arterial stiffness assessment parameter - Pulse Arrival Time (for palm and feet);

− Environmental parameters (temperature and relative humidity).
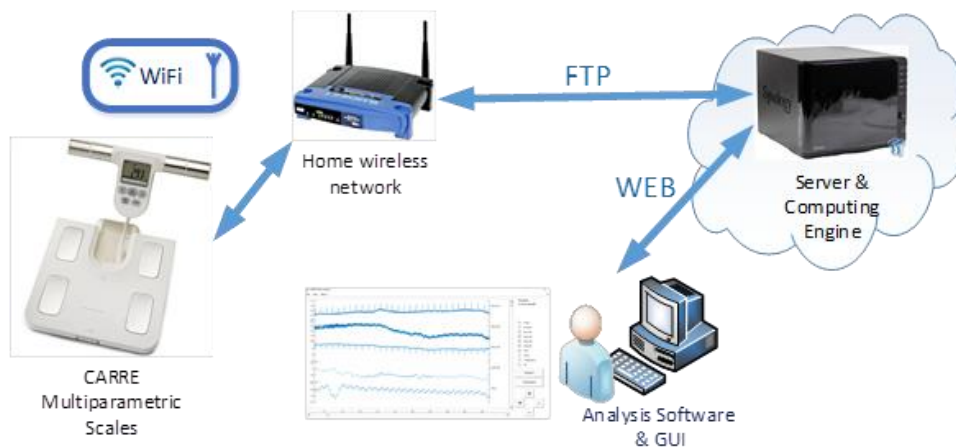
Figure 44. Deployment of CARRE Multiparametric Weight Scales
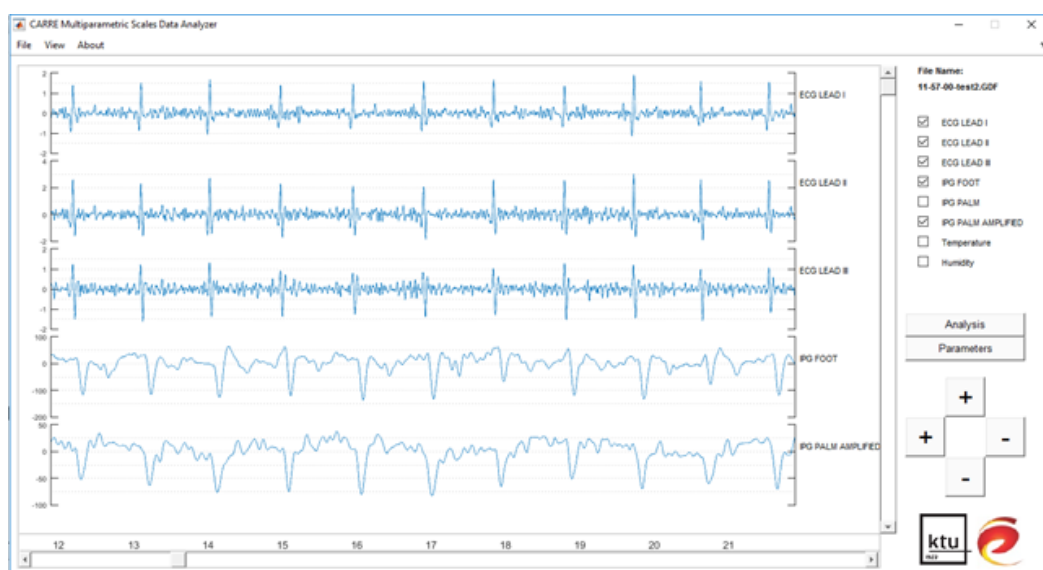
Figure 45. CARRE Multiparametric Weight Scales Data Analysis tool for preview and automated analysis

7.5. Risk factor entry, DSS alerts entry and educational resources aggregator server

The risk factor entry system, DSS alerts entry system and educational resources aggregator are three relatively independent subsystems which can be deployed separately. At this stage, they are in the same server for ease of deployment and management as well as for saving computer resources.

7.5.1. Server requirements

The hardware and software requirements for the server of risk factor entry system, DSS alerts entry system and educational resources aggregator are listed in the following table:

Table 6. Server requirements for risk factors entry system and education resources aggregator.	
Hardware requirements	
CPU	2 Cores, > 1.7 GHz (or more)
Memory	4GB (or more)
Hard Drive	150 GB (or more)
Software requirements	
Operating System	Linux, Ubuntu 14.04 LTS
Server	Apache 2.x server, NodeJS 4.x application server
Database	Redis (In Memory database)
Other	

7.5.2. General steps of the deployment process

7.5.2.1. Basic requirements

- Ubuntu 14.04 LTS Fresh installation
- SSH user with key-based authentication setup

7.5.2.2. Prepare dependencies

```
# install nodejs stuff
curl -sL https://deb.nodesource.com/setup_4.x | sudo -E bash –
sudo apt-get install -y nodejs
npm install –g grunt-cli gulp bower pm2


# install apache as a proxy/static server
apt-get install libapache2-mod-proxy-html
apt-get install libxml2-dev
```

```
a2enmod proxy && a2enmod proxy_ajp && a2enmod proxy_http && a2enmod rewrite &&
a2enmod deflate && a2enmod headers && a2enmod proxy_balancer && a2enmod
proxy_connect proxy_html
```

7.5.2.3. Installation setup for educational resources aggregator

```
git clone https://github.com/telemed-duth/carre-edu.git && cd carre-edu
npm install && bower install
grunt build
pm2 start production.json


# add it to apache
nano /etc/apache/sites-available/com.gr.educational.aggregator.conf


# COPY PASTE
<VirtualHost *:80>
    ServerAdmin admin@yourdomain.com
    ServerName edu.gr.carre-project.eu
    ProxyRequests off
    <Proxy *>
        Order deny,allow
        # modify the port here
        Allow from all://localhost:3010/
    </Proxy>
    <Location />
        PassReverse http://localhost: 3010/
        ProxyPassReverse http://localhost: 3010
    <Location />
</VirtualHost>


a2ensite com.gr.educational.aggregator.conf
apache2 reload
```

7.5.2.4. Installation setup for risk factors entry system

```
git clone https://github.com/carre-project/carre-entry-system.git && cd carre-
  entry-system
npm install && bower install
gulp build
pm2 start production.json


# add it to apache
nano /etc/apache/sites-available/com.gr.entry.system.conf
# COPY PASTE
<VirtualHost *:80>
    ServerAdmin admin@yourdomain.com
    ServerName edu.gr.carre-project.eu
    ProxyRequests off
    <Proxy *>
```

```
        Order deny,allow
        # modify the port here
        Allow from all://localhost:3020/
    </Proxy>
    <Location />
        PassReverse http://localhost:3020/
        ProxyPassReverse http://localhost:3020
    <Location />
</VirtualHost>


a2ensite com.gr.entry.system.conf
apache2 reload
```

7.5.2.5. Installation setup for SSL certificate

```
wget https://examples.carre-project.eu/letsencrypt/carre_certificates
#Run it twice, one for educational resources aggregator and one for risk factor
  entry system


sudo bash carre_certificates
```

7.6. System testing

The CARRE system is composed of coordinated components. Consequently the testing of the system needs to be carried out on the component level and the system level. During the design and implementation of each component, testing of individual components was carried out and reported in the corresponding deliverable. However, the integrative testing which involves extensive end user interactions is best performed at the evaluation stage by the patients and doctors; it will be reported in deliverable D7.4 "CARRE Evaluation & Implications Report".

8. Conclusions

This deliverable has reported the integration tasks of the CARRE system. The individual components that were designed according to the CARRE architecture have been implemented by partners, deployed at relevant partners' sites and tested in various ways. The details of each component were reported in previous deliverables.

The integration task ensures that the components of the system come together in a single executable system. The integrated system is exposed using standards-based RESTful web services, with which the user interface will then communicate. This will allow for computer–computer interaction with the service for future use and extension by others outside of the project. The integrated system will be given to users other than the developers (i.e. members of the use case partner teams) to test the system thoroughly.

Annex 1

RDF translations

1. Risk element names

English	Greek	Lithuanian
risk:has_risk_element_name	**risk:has_risk_element_name**	**risk:has_risk_element_name**
acute kidney disease	οξεία νεφρικη βλάβη	ūminis inkstų pažeidimas
acute myocardial infarction	οξύ έμφραγμα του μυοκαρδίου	ūminis miokardo infarktas
age	ηλικία	amžius
albuminuria	αλβουμινουρία	albuminurija
anemia	αναιμία	anemija
asthma	άσθμα	astma
atrial fibrillation	κολπική μαρμαρυγή	prieširdžių virpėjimas
beta-blockers	β-αναστολείς	beta-blokatoriai
central obesity	κεντρική παχυσαρκία	centrinis nutukimas
cholelithiasis	χολολιθίαση	tulžies pūslės akmenligė
chronic kidney disease	χρόνια νεφρική νόσος	lėtinė inkstų liga
chronic kidney disease stage 5	χρόνια νεφρική ανεπάρκεια σταδίου 5	lėtinė inkstų liga 5 stadija
chronic obstructive pulmonary disease	χρόνια αποφρακτική πνευμονοπάθεια	lėtinė obstukcinė plaučių liga
colorectal cancer	καρκίνος του παχέος εντέρου	kolorektalinis vėžys
contrast agents	σκιαγραφικές ουσίες	kontrastinės medžiagos
death	θάνατος	mirtis
death due to cardiovascular disease	θάνατος καρδιαγγειακής αιτιολογίας	mirtis dėl kardiovaskulinės ligos
depression	κατάθλιψη	depresija
diabetes	διαβήτης	diabetas
diabetic nephropathy	διαβητική νεφροπάθεια	diabetinė nefropatija
diuretics	διουρητικά	diuretikai
dyslipidemia	δυσλιπιδαιμία	dislipidemija
gastric cardia cancer	καρκίνος καρδιακής μοίρας του στομάχου	skrandžio įskrandžio srities vėžys
gastric non-cardia cancer	καρκίνος μη καρδιακής μοίρας του στομάχου	skrandžio ne įskrandžio srities vėžys
heart failure	καρδιακή ανεπάρκεια	širdies nepakankamumas
high-density lipoprotein cholesterol serum concentration	υψηλής πυκνότητας λιποπρωτεϊνης χοληστερόλης ορού	didelio tankio lipoproteinų cholesterolio koncentracija serume
hospitalization	νοσηλεία	hospitalizacija
hyperkalemia	υπερκαλιαιμία	hiperkalemija
hypertension	υπέρταση	hipertenzija
hyperuricemia	υπερουριχαιμία	hiperurikemija
hypoglycaemia	υπογλυκαιμία	hipoglikemija
hypotension	υπόταση	hipotenzija
ischemic heart disease	ισχαιμική καρδιοπάθεια	išeminė širdies liga
ischemic heart disease family history	οικογειακό ιστορικό ισχαιμικής καρδιοπάθειας	šeiminė išeminės širdies ligos anamnezė
ischemic heart disease self history	ατομικό ιστορικό ισχαιμικής καρδιοπάθειας	išeminė širdies liga anamnezėje

ischemic stroke	ισχαιμικό εγκεφαλικό επεισόδιο	išeminis insultas
left ventricular hypertrophy	υπερτροφία της αριστερής κοιλίας	kairiojo skilvelio hipertrofija
obesity	παχυσαρκία	nutukimas
obstructive sleep apnea	αποφρακτική υπνική άπνοια	obstrukcinė miego apnėja
osteoarthritis	οστεοαρθρίτιδα	osteoartritas
pancreatic cancer	καρκίνος του παγκρέατος	kasos vėžys
peripheral arterial disease	περιφερική αρτηριακή νόσος	periferinių arterijų liga
peripheral vascular disease	περιφερική αγγειακή νόσος	periferinių kraujagyslių liga
physical activity	φυσική δραστηριότητα/άσκηση	fizinis aktyvumas
renin-angiotensin system dual blockade	διπλός αποκλεισμός του άξονα ρενίνης-αγγειοτενσίνης	renino-angiotensino sistemos dviguba blokada
smoking	κάπνισμα	rūkymas
statins	στατίνες	statinai
triglycerides serum concentration	συγκέντρωση τριγλυκεριδίων ορού	trigliceridų koncentracija serume
cardiovascular events group 1	καρδιαγγεικά επισόδεια ομάδας 1	kardiovaskuliniai įvykiai 1 grupė
cardiovascular events group 2	καρδιαγγεικά επισόδεια ομάδας 2	kardiovaskuliniai įvykiai 2 grupė
cardiovascular events group 3	καρδιαγγεικά επισόδεια ομάδας 3	kardiovaskuliniai įvykiai 3 grupė
cardiovascular events group 4	καρδιαγγεικά επισόδεια ομάδας 4	kardiovaskuliniai įvykiai 4 grupė
cardiovascular events group 5	καρδιαγγεικά επισόδεια ομάδας 5	kardiovaskuliniai įvykiai 5 grupė

2. Observable names

English	Greek	Lithuanian
risk:has_observable_name	**risk:has_observable_name**	**risk:has_observable_name**
2h glucose after oral glucose tolerance test (mg/dL)	επίπεδα γλυκόζης 2 ώρες μετά το τεστ ανοχής χορήγησης γλυκόζης δια στόματος (mg/dL)	glikemija po 2 val. atliekant gliukozės toletavimo tyrimą (mg/dl)
2h glucose after oral glucose tolerance test (mmol/L)	επίπεδα γλυκόζης 2 ώρες μετά το τεστ ανοχής χορήγησης γλυκόζης δια στόματος (mmol/L)	glikemija po 2 val. atliekant gliukozės toletavimo tyrimą (mmol/l)
acute kidney disease diagnosis	διάγνωση οξείας νεφρικής ανεπάρκειας	ūminio inkstų pažeidimo diagnozė
acute myocardial infarction diagnosis	διαγνωση οξέος εμφράγματος του μυοκαρδίου	ūminio miokardo infarkto diagnozė
age (days)	ηλικία (μέρες)	amžius (dienomis)
age (months)	ηλικία (μήνες)	amžius (mėnesiais)
age (years)	ηλικία (έτη)	amžius (metais)
albuminuria	αλβουμινουρία	albuminurija
anemia diagnosis	διάγνωση αναιμίας	anemijos diagnozė
angina pectoris diagnosis	διάγνωση στηθάγχης	krūtinės anginos diagnozė
ankle-branchial index	κνημοβραχιόνιος δείκτης	kulkšnies - žasto indeksas
apnoea-hypopnea index	δείκτης υπνικής άπνοιας	apnėjos-hipopnėjos indeksas
asthma diagnosis	διάγνωση άσθματος	astmos diagnozė
atrial fibrillation diagnosis	διάγνωση κολπικής μαρμαρυγής	prieširdžių virpėjimo diagnozė
beta-blockers administration	χορήγηση β-αναστολέων	beta-blokatorių skyrimas

body fat	σωματικό λίπος	kūno riebalai
body mass index	δείκτης μάζας σώματος	kūno masės indeksas
cholelithiasis diagnosis	διάγνωση χολολιθίασης	tulžies pūslės akmenligės diagnozė
chronic kidney disease diagnosis	διάγνωση χρόνιας νεφρικής νόσου	lėtinės inkstų ligos diagnozė
chronic kidney disease diagnosis (5 levels)	διάγνωση χρόνιας νεφρικής νόσου (5 επίπεδα)	lėtinės inkstų ligos diagnozė (5 stadijos)
chronic kidney disease diagnosis (5 levels, sublevels)	διάγνωση χρόνιας νεφρικής νόσου (5 επίπεδα, υπο-επίπεδα)	lėtinės inkstų ligos diagnozė (5 stadijos, substadijos)
chronic obstructive pulmonary disease diagnosis	διάγνωση χρόνιας αποφρακτικής πνευμονοπάθειας	lėtinės obstrukcinės plaučių ligos diagnozė
colorectal cancer diagnosis	διάγνωση καρκίνου του παχέος εντέρου	kolorektalinio vėžio diagnozė
contrast agent administration	χορήγηση σκιαγραφικών ουσιών	kontrastinių medžiagų skyrimas
death diagnosis	διάγνωση θανάτου	mirties diagnozė
depression diagnosis	διάγνωση κατάθλιψης	depresijos diagnozė
diabetes diagnosis	διάγνωση διαβήτη	diabeto diagnozė
diabetic nephropathy diagnosis	διάγνωση διαβητικής νεφροπάθειας	diabetinės nefropatijos diagnozė
diastolic blood pressure	διαστολική πίεση	diastolinis kraujospūdis
diuretics administration	χορήγηση διουρητικών	diuretikų skyrimas
estimated glomerular filtration rate	εκτιμώμενος ρυθμός σπειραματικής διήθησης	apskaičiuotas glomerulų filtracijos greitis
fasting plasma glucose (mg/dL)	γλυκόζη νηστείας πλάσματος (mg/dL)	gliukozės koncentracija plazmoje nevalgius (mg/dl)
fasting plasma glucose (mmol/L)	γλυκόζη νηστείας πλάσματος (mmol/L)	gliukozės koncentracija plazmoje nevalgius (mmol/l)
gastric cardia cancer diagnosis	διάγνωση καρκίνου καρδιακής μοίρας του στομάχου	skrandžio įskrandžio srities vėžio diagnozė
gastric non-cardia cancer diagnosis	διάγνωση καρκίνου μη καρδιακής μοίρας του στομάχου	skrandžio ne įskrandžio srities vėžio diagnozė
glycated haemoglobin	γλυκοζυλιωμένη αιμοσφαιρίνη	glikuotas hemoglobinas
haemoglobin	αιμοσφαιρίνη	hemoglobinas
heart failure diagnosis	διάγνωση καρδιακής ανεπάρκειας	širdies nepakankamumo diagnozė
high-density lipoprotein cholesterol (mg/dL)	υψηλής πυκνότητας λιποπρωτεΐνης χοληστερόλη (mg/dL)	didelio tankio lipoproteinų cholesterolis (mg/dl)
high-density lipoprotein cholesterol (mmol/L)	υψηλής πυκνότητας λιποπρωτεΐνης χοληστερόλη ορού (mmol/L)	didelio tankio lipoproteinų cholesrolis (mmol/l)
hospitalization	νοσηλεία	hospitalizacija
hypertension diagnosis	διάγνωση υπέρτασης	hipertenzijos diagnozė
ischemic heart disease diagnosis	διάγνωση ισχαιμικής καρδιοπάθειας	išeminės širdies ligos diagnozė
ischemic heart disease family history	οικογειακό ιστορικό ισχαιμικής καρδιοπάθειας	šeiminė išeminės širdies ligos anamnezė
ischemic heart disease self history	ατομικό ιστορικό ισχαιμικής καρδιοπάθειας	išeminė širdies liga anamnezėje
ischemic stroke diagnosis	διάγνωση ισχαιμικού εγκεφαλικού επεισοδίου	išeminio insulto diagnozė
left ventricular hypertrophy diagnosis	διάγνωση υπετροφίας της αριστερής κοιλίας	kairiojo skilvelio hipertrofijos diagnozė

low-density lipoprotein cholesterol (mg/dL)	χαμηλής πυκνότητας λιποπρωτεΐνης χοληστερόλη ορού(mg/dL)	mažo tankio lipoproteinų cholesterolis (mg/dl)
low-density lipoprotein cholesterol (mmol/L)	χαμηλής πυκνότητας λιποπρωτεΐνης χοληστερόλη ορού (mmol/L)	mažo tankio lipoproteinų cholesterolis (mmol/l)
myocardial infarction self history	ιστορικό εμφράγματος του μυοκαρδίου	miokardo infarktas anamnezėje
non high-density lipoprotein cholesterol (mg/dL)	μη υψηλής πυκνότητας λιποπρωτεΐνης χοληστερόλη (mg/dL)	nedidelio tankio lipoproteinų cholesterolis (mg/dl)
non high-density lipoprotein cholesterol (mmol/L)	μη υψηλής πυκνότητας λιποπρωτεΐνης χοληστερόλη (mmol/L)	nedidelio tankio lipoproteinų cholesterolis (mmol/l)
obstructive sleep apnea diagnosis	διάγνωση αποφρακτικής υπνικής άπνοιας	obstrukcinės miego apnėjos diagnozė
obstructive sleep apnea diagnosis (3 levels)	διάγνωση αποφρακτικής υπνικής άπνοιας (3 επίπεδα)	obstrukcinės miego apnėjos diagnozė (3 stadijos)
osteoarthritis diagnosis	διάγνωση οστεοαρθίτιδας	osteoartrito diagnozė
pancreatic cancer diagnosis	διάγνωση καρκίνου του παγκρέατος	kasos vėžio diagnozė
peripheral arterial disease diagnosis	διάγνωση περιφερικής αρτηριακής νόσου	periferinių arterijų ligos diagnozė
physical activity	φυσική δραστηριότητα	fizinis aktyvumas
renin-angiotensin system dual blockade administration	χορήγηση διπλού αποκλεισμού του άξονα ρενίνης-αγγειοτενσίνης	renino-angiotensino sistemos dvigubos blokados skyrimas
serum creatinine (mg/dL)	κρεατινίνη ορού (mg/dL)	serumo kreatininas (mg/dl)
serum creatinine (micromol/L)	κρεατινίνη ορού (micromol/L)	serumo kreatininas (mkmol/l)
serum potassium (mEq/L)	κάλιο ορού (mEq/L)	serumo kalis (meq/l)
serum potassium (mmol/L)	κάλιο ορού (mmol/dL)	serumo kalis (mmol/l)
sex	φύλο	lytis
smoking status	κάπνισμα	rūkymas
statin administration	χορήγηση στατινών	statinų skyrimas
statin administration (list)	χορήγηση στατινών (λίστα)	statinų skyrimas (sąrašas)
systolic blood pressure	συστολική πίεση	sistolinis kraujospūdis
time after acute myocardial infarction event (days)	χρονικό διάστημα μετά από οξύ έμφραγμα του μυοκαρδίου (μέρες)	laikas po ūminio miokardo infarkto (dienomis)
time after acute myocardial infarction event (months)	χρονικό διάστημα μετά από οξύ έμφραγμα του μυοκαρδίου (μήνες)	laikas po ūminio miokardo infarkto (mėnesiais)
tobacco consumption (cigs. per day)	κατανάλωση καπνού (τσιγάρα/ημέρα)	tabako vartojimas (cig./dieną)
tobacco consumption (no, yes)	κατανάλωση καπνού (ναι/όχι)	tabako vartojimas (ne/taip)
total cholesterol (mmol/L)	ολική χοληστερόλη (mmol/L)	bendras cholesterolis (mmol/l)
total cholesterol (mg/dL)	ολική χοληστερόλη (mg/dL)	bendras cholesterolis (mg/dl)
triglycerides (mg/dL)	τριγλυκερίδια (mg/dL)	trigliceridai (mg/dl)
triglycerides (mmol/L)	τριγλυκερίδια (mmol/L)	trigliceridai (mmol/l)
uric acid serum concentration (mg/dL)	συγκέντρωση ουρικού οξέος ορού (mg/dL)	šlapimo rūgšties koncentracija serume (mg/dl)
uric acid serum concentration (mmol/L)	συγκέντρωση ουρικού οξέος ορού (mmol/L)	šlapimo rūgšties koncentracija serume (mmol/l)
urine output	παραγωγή ούρων	šlapimo išskyrimas

waist circumference	περιφέρεια μέσης	liemens apimtis
waist to height ratio	αναλογία περιφέρειας μέσης/ύψους	liemens - ūgio santykis
waist to hip ratio	αναλογία περιφέρειας μέσης/περιφέρεια γοφών	liemens -klubų apimties santykis
physical activity (steps)	φυσική δραστηριότητα (βήματα)	fizinis aktyvumas (žingsniai)
body weight	βάρος σώματος	kūno svoris
heart rate	καρδιακός παλμός	širdies susitraukimų dažnis
heart failure nyha classification	κατηγοριοποίηση nyha καρδιακής ανεπάρκειας	širdies nepakankamumo nyha klasifikaciją
physical activity (distance)	φυσική δραστηριότητα (απόσταση)	fizinis aktyvumas (atstumas)
body height	ύψος	ūgis
blood glucose (mg/dL)	σάκχαρο αίματος (mg/dL)	gliukozės vlood (mg/dL)
blood glucose (mmol/L)	σάκχαρο αίματος (mmol/L)	gliukozės vlood (mmol/L)

3. Measurement types

English	Greek	Lithuanian
risk: has_measurement_type_name	risk: has_measurement_type_name	risk:has_measurement_type_name
consumption	κατανάλωση	vartojimas
diagnosis (3 levels)	διάγνωση (3 επίπεδα)	diagnozė (3 stadijos)
diagnosis (5 levels)	διάγνωση (5 επίπεδα)	diagnozė (5 stadijos)
diagnosis (5 levels, sublevels)	διάγνωση (5 επίπεδα, υπο-επίπεδα)	diagnozė (5 stadijos, substadijos)
diagnosis (no, yes)	διάγνωση (όχι/ναι)	diagnozė (ne, taip)
distance (cm)	απόσταση (cm)	atstumas (cm)
drug administration	χορήγηση φαρμάκων	vaistų skyrimas
family history	οικογενειακό ιστορικό	šeiminė anamnezė
filtration rate	ρυθμός σπειραματικής διήθησης	filtracijos greitis
liquid output (ml/kg/h)	παραγωγή υγρού (ml/kg/h)	skysčių išskyrimas (ml/kg/val)
mass concentration (g/dl)	συγκέντρωση μάζας (g/dl)	masės koncentracija (g/dl)
mass concentration (mg/dl)	συγκέντρωση μάζας (mg/dl)	masės koncentracija (mg/dl)
mass over time (mg/24h)	μάζα στην πάροδο του χρόνου (mg/24h)	masė per tam tikrą laiką (mg/24val)
mass per height squared	μάζα ανά τετράγωνο ύψους	masė ūgio kvadratui
molar concentration (mEq/L)	μοριακή συγκέντρωση (mEq/L)	molinė koncentracija (meq/l)
molar concentration (mmol/L)	μοριακή συγκέντρωση (mmol/L)	molinė koncentracija (mmol/l)
molar concentration (μmol/L)	μοριακή συγκέντρωση (μmol/L)	molinė koncentracija (qmol/l)
no or yes	όχι ή ναι	ne ar taip
percentage	ποσοστό	procentai
physical activity	φυσική δραστηριότητα	fizinis aktyvumas
pressure (mmHg)	πίεση (mmHg)	spaudimas (mmhg)
rate (counts per day)	ποσοστό (μετρήσεις ανά μέρα)	dažnis (kiekis per dieną)

rate (counts per hours of sleep)	ποσοστό (μετρήσεις ανά ώρες ύπνου)	dažnis (kiekis per miego val.)
rate (counts per min)	ποσοστό (μετρήσεις ανά λεπτό)	dažnis (kiekis per min.)
ratio	αναλογία	santykis
sex (female, male, other)	φύλο (θήλυ, άρρεν, άλλο)	lytis (moteris, vyras, kita)
smoking (never, smoker, ex-smoker)	κάπνισμα (ποτέ, καπνιστής, πρώην-καπνιστής)	rūkymas (niekada, rūkantis, anksčiau rūkęs)
statin administration (list)	χορήγηση στατινών (λίστα)	statinų skyrimas (sąrašas)
time (days)	χρόνος (ημέρες)	laikas (dienomis)
time (months)	χρόνος (μήνες)	laikas (mėnesiais)
time (years)	χρόνος (έτη)	laikas (metais)
count	μέτρηση	kiekis
mass (Kg)	μάζα (Kg)	svoris (kg)
nyha classification	nyha κατηγοριοποίηση	nyha klasifikaciją
distance (Km)	απόσταση (Km)	atstumas (Km)

English	Greek	Lithuanian
risk:has_enumeration_values	**risk:has_enumeration_values**	**risk:has_enumeration_values**
no;mild;moderate;severe	όχι;ήπια;μέτρια;σοβαρή	nėra;lengvas;vidutinis;sunkus
no;stage1;stage2;stage3;stage4;stage5	όχι;επίπεδο1;επίπεδο2;επίπεδο3;επίπεδο4;επίπεδο5	nėra;1stadija;2stadija;3stadija;4stadija;5stadija
no;stage1;stage2;stage3A;stage3B;stage4;stage5	όχι;επίπεδο1;επίπεδο2;επίπεδο3A;επίπεδο3B;επίπεδο4;επίπεδο5	nėra;1stadija;2stadija;3Astadija;3Bstadija;4stadija;5stadija
no;yes	όχι;ναι	
no;yes		ne;taip
maternal;paternal;other	μητρικό;πατρικό;άλλο	motinos;tėvo;kita
no;yes	όχι;ναι	ne;taip
no;low;moderate;high	όχι;χαμηλή;μέτρια;υψηλή	nėra;mažas;vidutinis;didelis
female;male;other	θήλυ;άρρεν;άλλο	moteris;vyras;kita
never;smoker;ex-smoker	ποτέ;καπνιστής;πρωην-καπνιστής	niekada;rūkantis;anksčiau-rūkęs
atorvastatin;simvastatin;rosuvastatin;pravastatin;lovastatin	ατορβαστατίνη;σιμβαστατίνη;ροσουβαστίνη;πραβαστατίνη;λοβοστατίνη	atorvastatinas;simvastatinas;rozuvastatinas;pravastatinas;lovastatinas
no;classI;classII;classIII;classIV	όχι;κλάσηI;κλάσηII;κλάσηIII;κλάσηIV	nėra;Iklasė;IIklasė;IIIklasė;Ivklasė

English	Greek	Lithuanian
risk:has_label	**risk:has_label**	**risk:has_label**
g/day	g/ημέρα	g/dieną
cm	cm	cm

ml/min/1.73 m^2	ml/min/1.73 m^2	ml/min/1.73 m^2
ml/kg.h	ml/kg.h	ml/kg.h
g/dL	g/dL	g/dL
mg/dl	mg/dl	mg/dl
mg/24h	mg/24h	mg/24h
Kg/m^2	Kg/m^2	Kg/m^2
mEq/L	mEq/L	mEq/L
mmol/L	mmol/L	mmol/L
μmol/L	μmol/L	μmol/L
%	%	%
mmHg	mmHg	mmHg
counts/day	μετρήσεις/ημέρα	kiekis/dieną
counts/hours of sleep	μετρήσεις/ώρες ύπνου	kiekis/miego valandos
counts/min	μετρήσεις/λεπτό	kiekis/minutę
days	ημέρες	dienos
months	μήνες	mėnesiai
years	έτη	metai
counts	μετρήσεις	skaičius
Kg	Kg	Kg
Km	Km	Km

Annex 2

Software of Integrated CARRE Environment

What is CARRE Integrated Service Environment?

The CARRE integrated services is an aggregation of subsystems that each is responsible for different type of functions. The integration procedure involves 4 Virtual machine compilations including all the required configurations and interlinking between them in order to function as a whole.

The VM configuration phase is described in D7.3 section 7.

Download

For security measures the CARRE Virtual Machines (VMs) are not available for unauthenticated users.

- VM1 – Visual interface and medical evidence aggregator
- VM2 – Authentication, data aggregation, decision support and storage server
- VM3 – PHR manual entry system server
- VM4 – Risk factor entry, DSS alerts entry and educational resources aggregator server

The CARRE Integrated Service Environemnt is Open Source

CARRE Integrated Service Environment can be freely used in Open Source applications under the terms GNU General Public License (GPL).

Copyright © 2016, CARRE Project