BD2Decide

Big Data and models for personalized Head and Neck Cancer Decision support

TITLE	**System architecture**		
Deliverable No.	D2.3		
EDITOR	Alessio Fioravanti (UPM), Liss Hernández (UPM), Laura Lopez (UPM), Giuseppe Fico (UPM), Maria Teresa Arredondo (UPM)		
Contributors	Eleni Mantziou (ATC), Vasilis Tountopoulos (ATC), Leonidas Kallipolitis (ATC), Florian Jung (Fraunhofer), Stefan Wesarg (Fraunhofer), Thomas Klausch (VU/VUmc), Peter van de Ven (VU/VUmc), Franco Mercalli (MME), Ron Shefi (AII), Avner Algom (AII), Giacomo Feliciani (MAASTRO), Philippe Lambin (MAASTRO), Luca Mainardi (POLIMI), Valentina Corino (POLIMI), Eros Montin (POLIMI)		
WorkPackage No.	WP2	**WorkPackage Title**	Requirements
Status[1]	FINAL	**Version No.**	1.0
Dissemination level	PU		
DOCUMENT ID	D2.3 System architecture		
FILE ID	BD2Decide D2.3		
Related documents	D5.1 Multilayer data acquisition and management services		

[1] Status values: TOC, DRAFT, FINAL

Distribution List

Organization	Name of recipients
AOP	T. Poli, E.M. Silini, E. Martinelli, G. Chiari, C. Caminiti, G. Maglietta, D. Lanfranco
VU/VUmc	R. H. Brakenhoff, H. Berkhof, P. va de Ven, Th. Klausch
UDUS	K. Scheckenbach
INT	F. Favales, L. Licitra, E. Montini, G. Calareso, G. Gatta, A. Trama
UPM	G. Fico, A. Fioravanti, L. Hernández, L. Lopez, M.T. Arredondo
POLIMI	L. Mainardi, E. Montin, V. Corini
Fraunhofer	F. Jung, S. Wesarg
ATC	V. Tountopoulos, L. Kallipolitis, E. Mantziou
MAASTRO	P. Lambin, F. Hoebers, A. Berlanga, G. Feliciani, R. Leijenaar
AII	A. Algom, A. Kariv, R. Shefi
MME	S. Copelli, F. Mercalli
UNIPR	S. Rossi, M. Silva
European Commission	Project Officer: and all concerned E.C. appointed personnel and external experts

Revision History

Revision no.	Date of Issue	Author(s)	Brief Description of Change
0.1	28.10.2016	L. Hernández, A. Fioravanti, L. López, G. Fico	ToC
0.2	22.11.2016	L. Hernández, A. Fioravanti, L. López, G. Fico	ToC
0.3		L. Hernández, A. Fioravanti, L. López, G. Fico, F. Jung, V. Corino, V. Tountopoulos, L. Kallipolitis	Sections added: 1, 2, 2.4 (2.4.1), 3 (3.1), 4 (4.1), 5 (5.1), 6 (intro), 2.4.5, 3.5, 4.4, 5.4, 2.4.3, 4.3, 5.3
0.4		L. Hernández, A. Fioravanti, L. López, G. Fico	Sections added: Abstract and chapter 1 Sections modified: 3.2, 4.1

0.5	L. Hernández, A. Fioravanti, L. López, G. Fico, R. Shefi, F. Jung, V. Corino, L. Kallipolitis	Sections added: 2.4.4, 2.4.4.1, 3.5, 2.4.6, 3.7, 4.5, 2.4.2, 3.3, 4.3, 5.2, 5.3, 2.4.6, 3.7, 4.5, 5.5, 2.3.4, 2.4.4.2 Sections updated: 5.4
0.6	L. Hernández, A. Fioravanti, L. López, G. Fico, L. Kallipolitis	Sections added: 3.1, 6 Sections updated: all
1.0	L. Hernández, A. Fioravanti, L. López, G. Fico	Final version

Addressees of this document

This document is addressed to the BD2Decide Consortium and provides the technical description of the overall DB2Decide system architecture. More specifically, it includes the general architecture definition of the BD2Decide platform and a detailed definition of each BD2Decide system module and corresponding components, including relevant UML and data flow diagrams.

The deliverable also depicts a description of the interoperability protocols of the relevant standards to be used within the BD2Decide system. Moreover, the integration of the external data sources is presented.

The main result of this deliverable is the final definition of the system architecture including the global technical design and the integration requirements for the single components.

This document will be delivered to the European Commission.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abbreviations and definitions

BD	Big Data
BDI	Big Data Infrastructure
CDSS	Clinical Decision Support System
DICOM	Digital Imaging and Communications in Medicine
DPEE	Digital Patient Exploration Environment
eCRF	Electronic Clinical Record Form
EHR	Electronic Health Record
FE	Feature Extractor
H&N	Head and Neck
H&NC	Head and Neck Cancer
KMS	Knowledge Management System
ICD	International Classification of Diseases
ICT	Information and Communication Technologies
IdM	Identity Management
IPDA	Interactive Patient Decision Aid
IPDAS Collaboration	International Patient Decision Aid Standards Collaboration
IPRR	Integrated Patients' Records Repository
LOINC	Logical Observation Identifiers Names and Codes
MEDLINE	Medical Literature Analysis and Retrieval System Online
NAS	Network-attached storage
PDS	Patient Documentation System
PT	Phenotypization Tool
QoL	Quality of Life
ROI	Region of Interest
SNOMED	Systematized Nomenclature of Medicine
TBCE	Tumor Board Collaboration Environment

UCD	User Centric Design
UI	User Interface(s)
UML	Unified Modelling Language
VAT	Visual Analytics Tool

Abstract

The purpose of this document is to describe the architecture of the BD2Decide system, along with the presentation of the different system components, including the data repositories, the user interfaces and the analysis models. UML and data flow diagrams are presented for each architecture module.

Moreover, this deliverable presents a first concept of the integration of the external data sources and the technologies of the BD2Decide system.

1. ABOUT THIS DOCUMENT

1.1 Introduction and scope

This document defines the technical architecture and the design of the BD2Decide platform and of its components, the data sharing and interoperability protocols and standards required for usage in clinical settings, according to the proposed design.

The deliverable also defines the baseline sizing, archiving and performance requirements, by identifying the required hardware and software specifications.

The document relies on the results of "D2.1 User needs and use cases", "D2.2 User interaction sketches" and "D5.1 Multilayer data acquisition and management services".

1.2 Structure of the deliverable

Following the objectives set for this deliverable, the document is structured as follows:

- Section 2 depicts a detailed view and description of the overall BD2Decide architecture.
- Section 3 shows the BD2Decide technologies, including the description of the principal system tools: Visual Analytics tool, Interactive Patient's co-Decision Aid tool, Clinical DSS tool, Big Data Infrastructure, Data semantic and data linking, Knowledge Management System with the related ontology and finally the imaging and statistical models.
- Section 4 shows the UML diagrams of the system modules.
- Section 5 shows the data flow diagram of each module.
- Section 6 is dedicated to the integration of the external data sources.

2. BD2DECIDE ARCHITECTURE

The chapter illustrate the BD2Decide system architecture, the system' components specifications and their role in the overall architecture, in order to provide functional specifications that will drive their implementation.

2.3 BD2Decide system architecture

A high level representation of the system architecture, presenting the main modules and components, is provided in Figure 1.

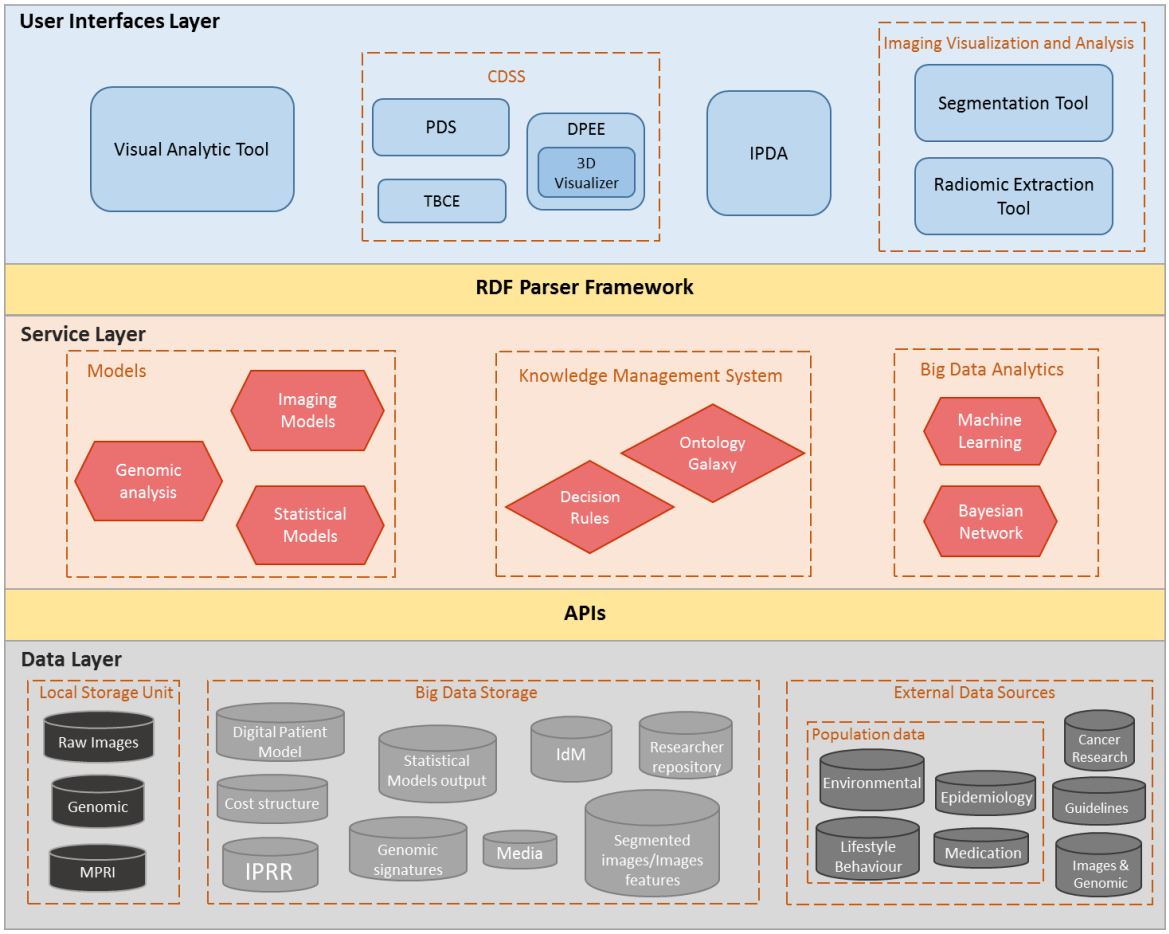


Figure 1 - BD2Decide System Architecture

The architecture is represented following a three tier approach: User Interfaces layer, Models and Service layer and Data layer. An RDF Parser Framework is responsible of allowing the applications to use the services. The APIs give the applications access and services access to the data. The elements of each layer and their relative functionalities and data exchange are described in the sections below.

2.3.1 User Interfaces Layer

This layer represent the part of the architecture in charge of allowing the end-user interaction with the BD2Decide system. The layer is composed by the following modules:

- *CDSS:* it consists of the clinician interface. Clinicians will be able to aggregate new patients into the clinical registry, manage patient treatment in the form of assessments, care recommendations, alerts and reminders.
    - DPEE: Digital Patient Exploration Environment, aiming at reducing medical subjectivity and improving clinical interpretation at patient level through an interactive explorative 3D head/neck model as well as full body avatar. The module communicates with the *Digital Patient Model* database.
    - PDS UI: Patient Documentation System User Interface, interface used by clinicians for clinical activities such as therapeutics, preventive care, comprehensive disease management, diagnostics, image recognition and interpretation and prognostics.
    - TBCE: Tumor Board Collaboration Environment, supporting the multi-disciplinary decision making through a remote interaction
- *Visual Analytic Tool*: the tool assists the work of clinical researchers active in H&N cancer, by use of querying and aggregating data, considering trends and identifying patients' clusters.
- *Interactive Patient's co-Decision Aid (IPDA):* it presents patient's treatment alternatives as identified by the physician along with curative effectiveness and impacts/side effects. The IPDA makes use of the *Media* database information.
- *Imaging Visualization and Analysis*: the tools for image segmentation (Segmentation Tool) and radiomic features extraction (Radiomic Extraction Tool).

2.3.2 Service Layer

This layer gathers together the models to be used for data input analysis, the Knowledge Management System and the Big Data Analytics.

1. The Models are constituted of:
    a. *Imaging Models*, based on image- and model-based information technology i.e. image enhancement, segmentation and feature radiomics extraction process for H&N tumor and lymph-nodes.
    b. *Genomic Analysis*, responsible of genomic and biomolecular diagnostic tests (Illumina™, qRT-PCR equipment, etc.). The output of the genomic analysis is stored into the *Genomic signatures* database.
    c. *Statistical Models*, a library of prognostic models that takes into account heterogeneity between subpopulations in terms of their baseline risk. Inputs are coming from clinical data (*IPRR*) and *Population data*. Results from the module are stored into the *Statistical Model output* database.
2. *Knowledge Management System (KMS)*: The Knowledge Management System converts data into normalized and structured information and extract it into

knowledge. It makes use of semantic multiscale and multivariate data modelled through an ontology (Ontology Galaxy). A set of rules for extracting imaging features, improve the prognostic models' outcomes and finding statistical evidence from retrospective cases and from research are generated from the KMS.

3. *Big Data Analytics*: the module constitutes the core of advanced large data examination processes, sets to uncover hidden patterns, unknown correlations, medical treatments trends, clinical preferences and other useful clinical decision information. The module combines built-in statistical techniques for predictive analytics, data mining and text mining including machine learning and Bayesian Network approaches paradigms. A rule-based reasoning approach will allow the extraction of dataset features, with the intent of finding statistical evidence from retrospective cases and from research, thanks to the support of semantic querying (SPARQL/RDF).

2.3.3 Data Layer

The Data Layer is composed of three databases groups:

1. *Local Storage Unit*: it represents the local storage unit placed on each clinical center. The storage unit contains information that is not shared among the system's modules and that it is only accessed internally from each clinical center. A Network Attached Storage (NAS) will be adopter for the data storage and a computer for the dataset access.
    a. *Raw images*, responsible of storing the tumor cancer raw images and the corresponding metadata information. The images are previously anonymized and encoded. The raw images are used as inputs for the *Imaging Models* module,
    b. *Genomic*, responsible to store the BAM genomic files. This database is used as input for the *Genomic Analysis*.
    c. *MPRI*, responsible to store patients' personal information. This database matches the real name patient with the corresponding patient ID and with the associate clinical data stored into the *IPRR* module.
2. *Big Data Storage*: it contains the dataset used as inputs for the Big Data Infrastructure (see D6.1 for more details): it is composed of the following repositories:
    a. Statistical Models output: results from the prognostic statistical models.
    b. Digital Patient Model: results from the Virtual Patient tool.
    c. Researcher repository: database used for the storage of the Visual Analytics Tool research activities.
    d. Cost structure: results from the Cost-utility analysis tool.
    e. Media: videos and animations inputs used by the IPDA
    f. IdM: contain the system login users permission information
    g. IPRR: Integrated Patients' Records Repository containing all clinical information

   h. Genomic signatures: represent the result information coming from the BAM analysis.
   i. Segmented images and images features: output from segmentation and radiomic features tool.
3. *External Data Source*: this group of databases gather together information from H&NC patients general health status, education, census, cancer incidence and prevalence, cancer death, treatment, presence of pollutants, risk factors (smoking, alcohol use/abuse) in sub-populations patients' longitudinal data derived from data collection of the retrospective and prospective cohorts enrolled by BD2Decide clinical centers. External data sources management is detailed in the deliverable D7.2. The external data sources include inputs from:
   a. Population Data:
      i. Cancer Registries data: cancer registries data coming from:
         1. INT Cancer Registries
         2. External Cancer Registries
      ii. External Images and Genomic databases
      iii. Environmental, Epidemiology, Life Behavior and Medication Data coming from:
         1. Istituto Superiore Sanità (Population demographic data)
         2. External dataset
   b. Other External Data:
      i. Literature

Once introducing the data layer, in the following paragraph is described the management of all these data including the structure and location of each type of data.

2.3.4 Patients' data repository management

One of the main goal of the BD2Decide system is to explore the potential value of applying the cloud computing and Big Data techniques to the healthcare decision making process and more specific for the discovery and validation of personalized prognostic patterns for H&NC. The Big Data approach involves handling a large amount of data and the analytics techniques to request them in a decentralized way. In this paragraph the management of the BD2Decide dataset is presented including the structure and location of each data type.

The data included in the BD2Decide system are classified as following:

1. *Patient's Data*: this includes information of the electronic health record for BD2Decide patients. This information gathers demographic and clinical data, pathological, genomic and imaging data of patients. This also includes data relates with risk factors of developing a head and neck cancer. Other information collected is about patients' treatment, in particular surgery, chemotherapy and radiotherapy data, and the level of toxicity as a result of treatment. Also collects data of the follow-up period and questionnaires of quality of life from patients.

2. *Population Data*: this includes aggregated data from external sources related with smoking habits, alcohol use, co-medication, hospitalization, environmental exposures, cancer registries and behavioral data.
3. *Clinical Guidelines and Cancer Research*: this includes information about relevant scientific documentation, clinical and best practice guidelines and data related with treatments investigations and studies.

The BD2Decide system makes use of patients data collected from five clinical centers in Italy (INT and Parma), Netherlands (MAASTRO and VU/VUmc) and Germany (UDUS). Data related with patient will collected through OpenClinica tool and was defined in the internal document eCRF [See eCRF scheme (Annex I to D2.1)]. This data will be used for retrospective study and the same data will be gathered using the CDSS in the prospective study performed inside the project.

The Figure 2 shows the structure and location of the data repositories. BD2Decide's data repositories will be divided in local repositories located in each clinical centers and central repositories that will be included in the Big Data infrastructure, in particular located in the Storage Environment.

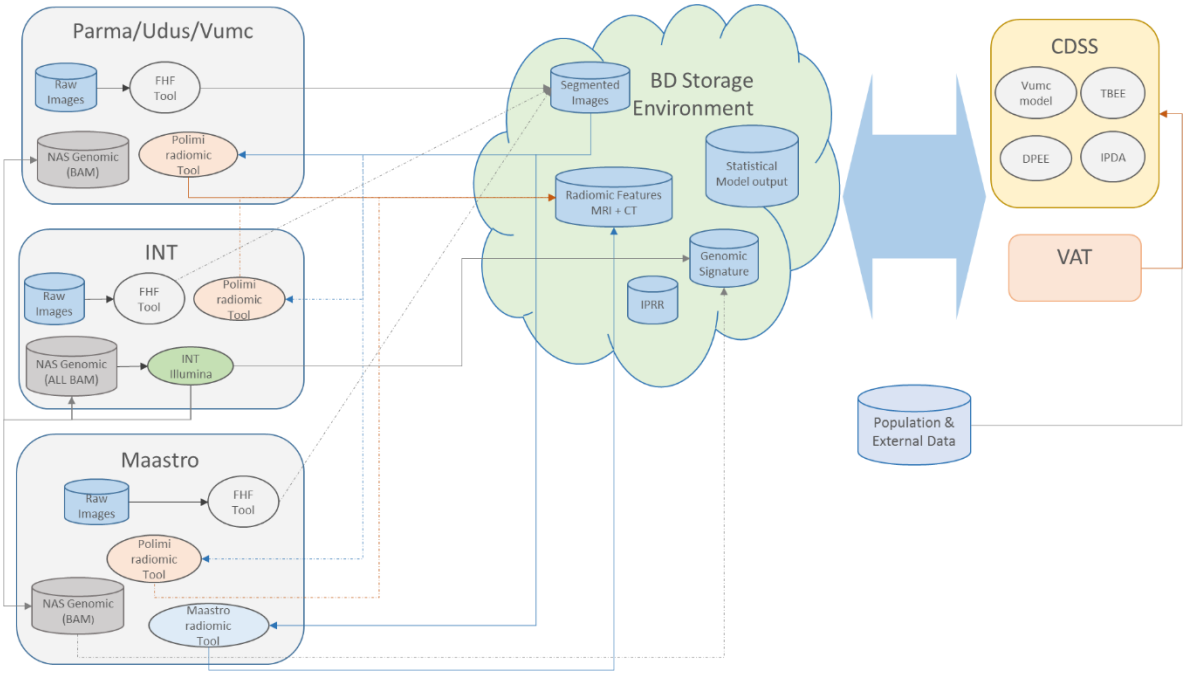


Figure 2 - BD2Decide datasets, service location and communication

Local repositories are located in each clinical centers and contains information related to the raw images from MRI and CT scans and BAM files with genomic data. These information is not shared with the system's modules and only is used by imaging processing and genomic processing tools installed locally.

The Storage Environment of Big Data Infrastructure contains the Segmented Images and Radiomic Features repositories that store the outcomes of imaging process performed locally in each center. The outcomes of genomic analysis are stored in the Genomic Signature repository that are also located in the BD Storage Environment. In this environment is also stored the outcomes of the statistical models (Statistical Model output repository) and in the IPRR are stored al clinical data collected from patients involved in BD2Decide's retrospective and prospective studies.

The population data and literature references are extracted from external data repositories that are requested through public APIs and this information is used by Clinical DSS and VAT tools.

2.4 Individual components of BD2Decide architecture

The paragraph describes in detail the main component of the BD2Decide system and the communication among them. These components are:

1. Models: prognostic models libraries, statistical methodologies, imaging prognostic models and genomics and biomolecular patient's-tumor characterization predictors.
2. Storage Environment: the whole set of units where the collection of data is stored.
3. External Data Sources: population-specific, epidemiological, behavioral and environmental data, clinical guidelines and cancer research registries
4. User Interfaces: the whole set of UIs allowing the control of the different units by end-users, the Visual Analytics Tool, the Clinical Decision Support System (CDSS) and the Interactive Patient's co-Decision Aid (IPDA).
5. Big Data Analytics and Infrastructure: it represents the multisource data analysis and extraction, supported by a knowledge management system able to homogenize and mine existing know-how from research, clinical trials and patient's clinical records.

2.4.1 Clinical DSS tool suite

The Clinical DSS tool suite will provide the facilities that will allow clinicians to manage the patients' data at the stages of diagnosis, treatment and follow-up as they have been defined in D2.2. The following list summarizes the major needs of clinicians that will be covered by the provided CDSS tools:

- Management of demographic data of the patients
- Analysis of pathological data
- Management of radiological analysis, including imaging analysis and radiomics generation
- Analysis of genomics
- Management of treatment, including prognostic models, patient questionnaires and online boards
- Management of follow-up actions

As analytically presented with the User Interface sketches in D2.2, the CDSS covers the above needs using a number of modules to handle the required user interaction with the BD2Decide system. The interactions of CDSS with the other architectural components are presented in Figure 3. The modules that are handling the interactions and the relevant involved BD2Decide components are analytically presented in the list right after it.

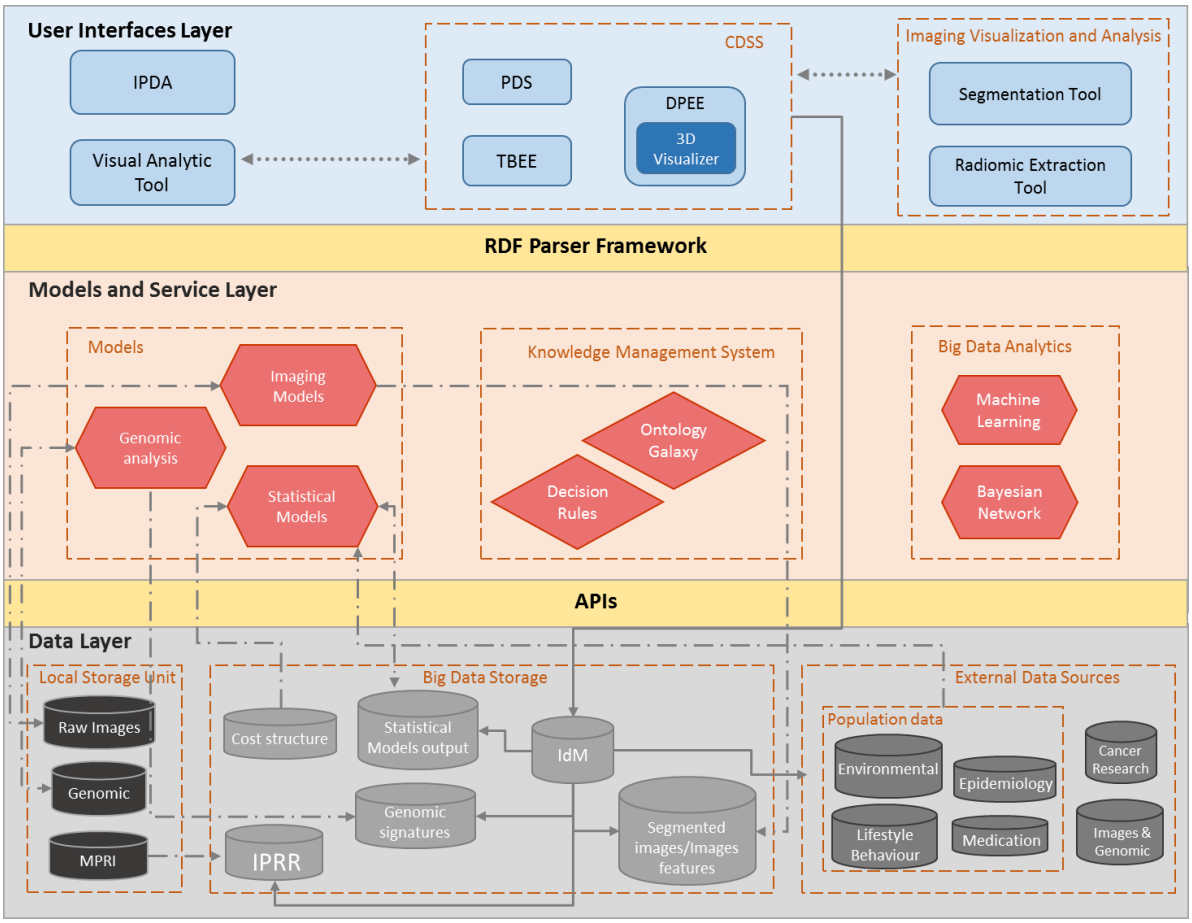


Figure 3 - CDSS Architecture

In the following figure (Figure 4) all the elements involved in the CDSS are grouped by type.

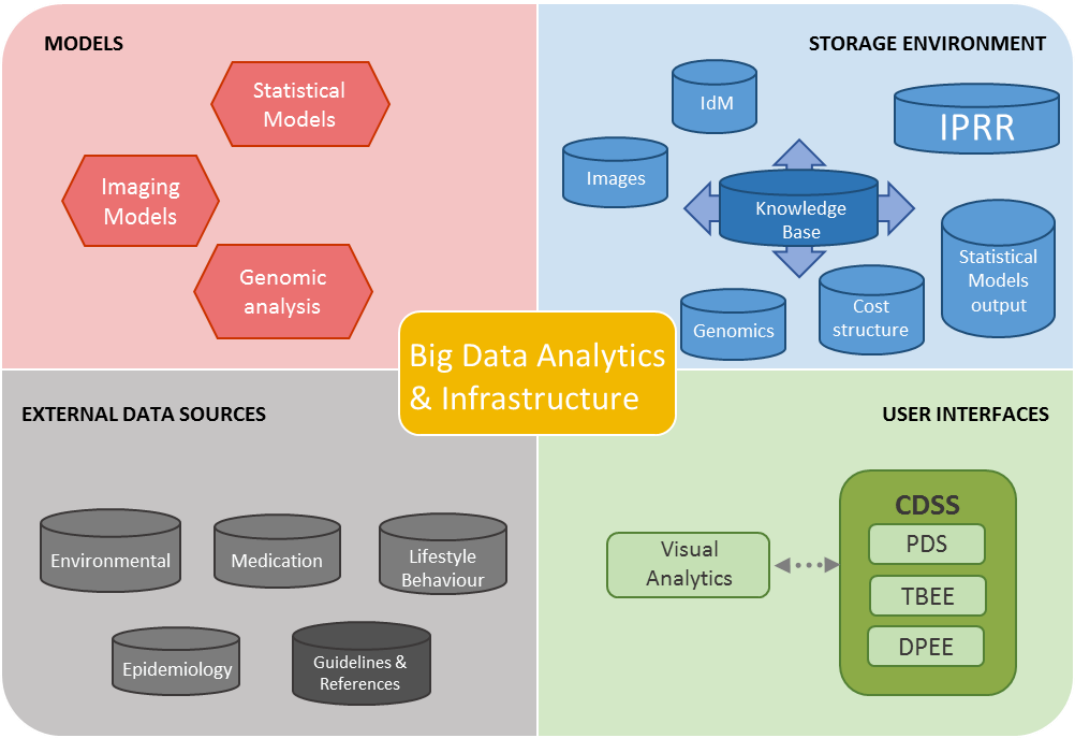


Figure 4 - CDSS architectural modules

- **The dashboard module,** which provides a unified environment presenting all the available functionalities to clinicians. It is actually the interface of the CDSS which includes links to all the underlying tools and visualizes information coming by other components of the system. It includes all the different data views of a patient, the grouping of data according to the treatment process stage and the links to the external tools that will have to be invoked in order to initiate various processes, e.g. image analysis. This is the module including all the interface aspects of the modules and tools described below.
  Components involved**:**
  - IdM
- **The eCRF management module,** which includes the user interface to manage data stored in the eCRF of patients as well as the necessary communication mechanisms to fetch and save the relevant data to the database (PDS).
  Components involved**:**
  - IPRR, Genomic signatures, Segmented images
- **The tumor localization tool,** which is a graphical environment allowing clinicians to localize the tumor location on a preloaded sketch using simple drawing options.
  Components involved**:**
  - IPRR
- **The DPEE UI,** which provides a visual representation of a patient's virtual model through a 3-D avatar model. This UI will allow a 360^{o} examination of the patient's model and help clinicians better understand the tumor characteristics.

Components involved**:**
  - IPRR, Segmented images
- **Prognostic Prediction Visualizations,** which offer a set of interactive graphs and charts that visualize the output coming from the prognostic prediction analysis tool. The visualizations are accompanied by a set of filters allowing the clinicians to exactly define the parameters of the model that is used by the analysis tool. This way, prediction results can be tailored to specific factors that can affect the patient's treatment process like e.g., place of living, age, etc. Moreover, differently defined models can be used simultaneously to generate predictions that can be compared on the same screen and help clinicians take decisions. The analysis tool will also provide data presenting clusters of patients based on statistical analysis which will be also visualized via configurable charts so as to help clinicians decide on the proper treatment method.
  Components involved**:**
  - IPRR, Statistical Models Output, Population data
- **Tumor Board Organizer,** which will be a tool offering the functionalities of setting up, running and keeping notes of online meetings between health professionals.
  Components involved**:**
  - IPRR
- **The QoL assessment tool,** which includes the infrastructure required to support the Quality of Life questionnaires that will be answered by the patients. The questionnaires are implemented as online forms. The results will be saved in the PDS and scores will be automatically calculated and displayed to the clinician through the CDSS. The QoL assessment tool will manage all this process by getting the patient's answers to the questionnaires, calculating and saving the scores, and finally visualizing them in the CDSS.
  Components involved**:**
  - IPRR

2.4.2 Visual Analytics Tool

This section covers the Visual and Presentation Suite and more specifically the Research scenario. This tool will include insightful and intuitive graphs, charts, and images that communicate risk information and offer effective decision support options.

This suite aims to create a visual analytics module focused on the exploitation, representation and visualization of information (e.g. patients' clinical cases, population-based information, etc.) retrieved from large-scale and heterogeneous sources.

Knowing the purpose of the Visual Analytics Tool, the Figure 5 represents the architecture scheme of it. In this diagram all the datatype and elements involved in the BD2Decide system which interact with the researcher tool are represented. The aim of this diagram is to

understand the interaction of the tool with the other BD2Decide system elements. In this way the dependencies and other relations are identified and specified.

The 'communication' arrows (continuous line) mean that the VAT can access (through All-In-Image APIs) to the data once the user credentials are authenticated. 'Direct access' arrows (·····) mean that there is a direct way (within the application) to connect with other applications (CDSS). Finally, 'indirect access' arrows (– – ·) represent those elements who take part of the tool but in an indirect way, for example, VAT uses the imaging models output, and because of that it is specified in the VAT architecture with the 'indirect access' link.

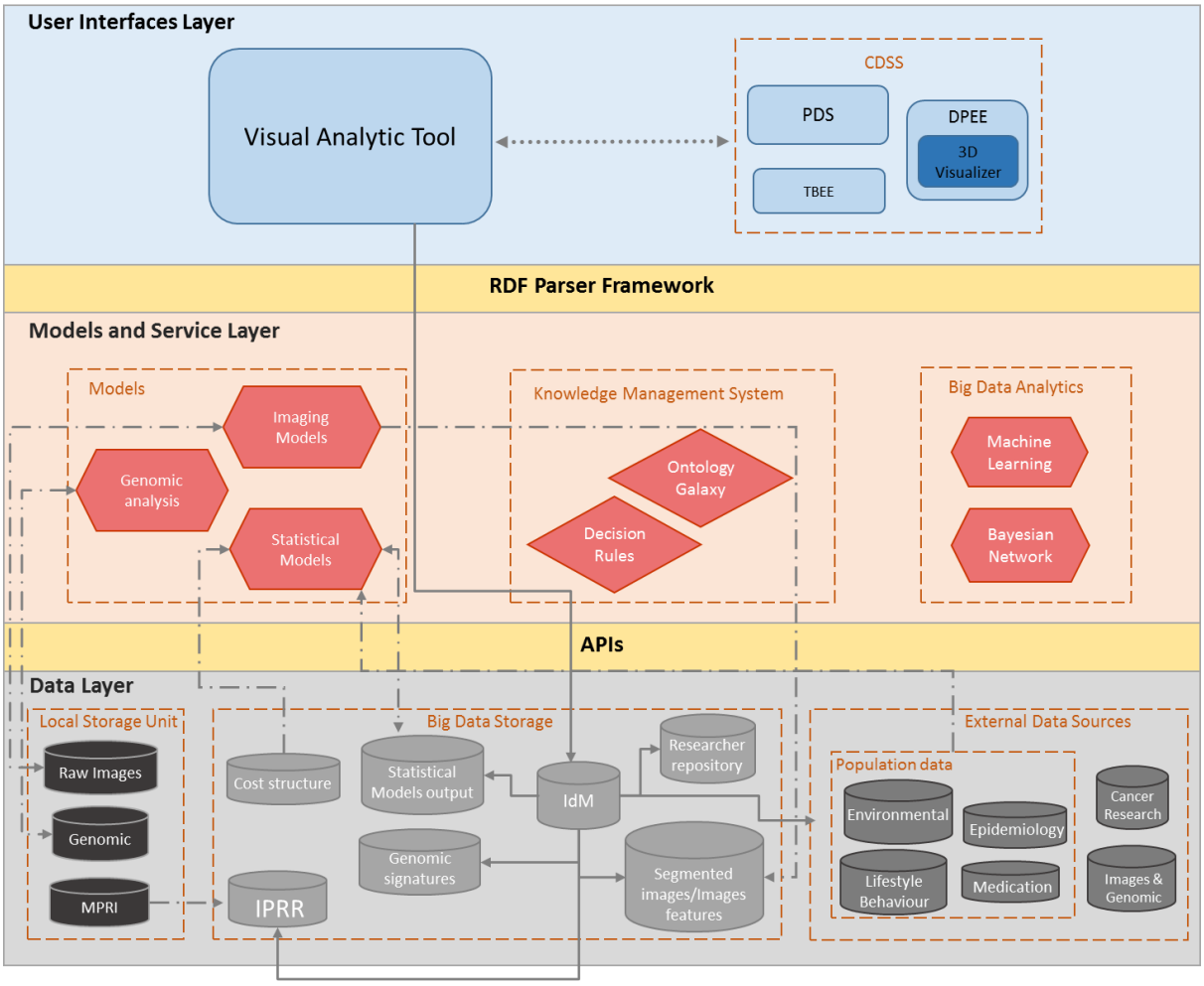


Figure 5 - Visual Analytics Tool architecture

In the following figure (Figure 6) all the elements involved in the VAT are grouped by type.

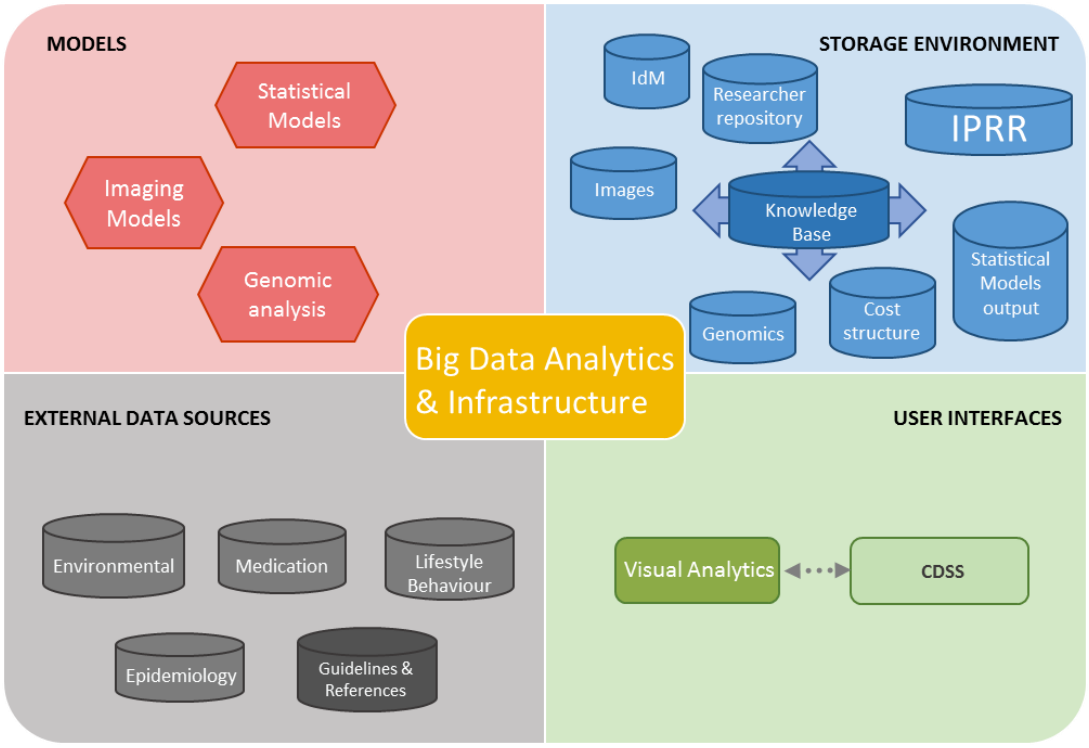


Figure 6 - VAT architectural modules

Each functionality of the VAT defined as of today, listed below, uses the corresponding components to achieve it purpose. The Ontology Galaxy is involved in each functionality.

Login:

- Function: access to the Visual Analytics Tool.
- Components involved:
  - Identity management database (IdM) [input].

Network profile:

- Function: specify the skills, field of interest and expertise of the researcher.
- Components involved:
  - Researcher repository [input and output].

Network activity:

- Function: see other researchers' works to be up to date and to see if some works can be helpful to complete the current researcher work.
- Components involved:
  - Researcher repository [input and output].

Projects (researchers' works):

- Function: virtual folders to contain each element of the researcher work: queries, literature searched, outcomes…
- Components involved:
  - Researcher repository [input and output].

Dashboard (H&NC overview):

- Function: give an overview of the H&NC status taking into account the data integrated in BD2Decide system.
- Components involved:
  - External data sources [input].
  - BD2Decide clinical data [input].
    - Images and genomics.
    - IPRR.
    - Statistical models output.
  - Researcher repository [input and output].
  - Big data analytics [input].

Data analysis & Compare similar cases:

- Function: exploit the BD2Decide databases, querying the data and customizing the results to obtain the outcomes needed by the researcher.
- Components involved:
  - BD2Decide clinical data [input].
    - Images and genomics.
    - IPRR.
    - Statistical models output.
  - Researcher repository [input and output].
  - Models [input].
    - Prognosis (statistical).
    - Imaging.
    - Genomic.
  - Big data analytics [input].
  - Decision rules [input].

External data access:

- Function: access to external datasets to obtain additional information about H&NC status and tend.
- Components involved:
  - External data sources [input].
    - Environmental.
    - Epidemiology.
    - Lifestyle and behavior.
    - Medication.

        - Imaging & genomic.
    - Researcher repository [input and output].
    - Big data analytics [input].

Decision maker:

- Function: give direct support to researcher including not only predictions about data but also cost information of adding further test or treatments.
- Components involved:
    - BD2Decide clinical data and cost structure [input].
        - Images and genomics.
        - IPRR.
        - Statistical models output.
        - Cost structure.
    - Researcher repository [input and output].
    - Models [input].
        - Prognosis (statistical).
        - Imaging.
        - Genomic.
    - Big data analytics [input].
    - Decision rules [input].

Literature (alerts and search):

- Function: access to literature related with the researcher work to give more information to the researcher.
- Components involved:
    - External data sources [input]:
        - Literature (such as PubMed).
    - Researcher repository [input and output].
    - Big data analytics [input].

User management:

- Function: manage all user permissions into the Visual Analytics Tool.
- Components involved:
    - IdM [input and output].

The following figure shows the architectural view of the VAT, including the relations between each module involved. To simplify the view *Data analysis & Compare similar cases*, *External data access* and *Decision maker* are grouped in *Data analysis* box. For more details related with the data flow diagram see the section 5.2.

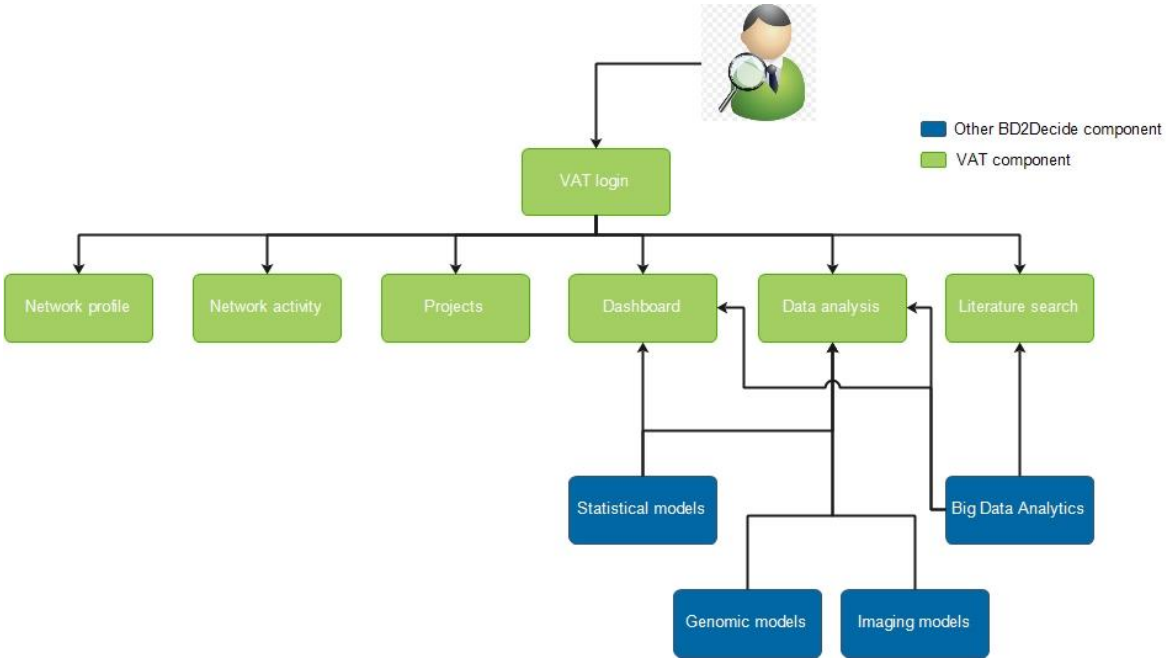


Figure 7 - VAT architecture view including VAT individual components

In the following sections all the models (2.4.4, 2.4.6) and data flow (4.2, 4.1 , 4.4, 4.5, 5.2, 5.4, 5.5) are detailed with the purpose of understanding the interactions and usage of the BD2Decide data and architecture.

2.4.3 Interactive Patient's co-Decision Aid

The Interactive Patient's co-Decision Aid (IPDA) is a tool that helps patients to become involved in Shared Decision Making by clarifying the treatment or medical decision that needs to be taken. It has to provide two major functionalities:

- Easy, visual information about treatment options based on multimedia content for the patients.
- Information about the patients' preferences and life habits that can help clinicians select a treatment method during consultation.

The following figure present an architectural view of the IPDA tool:

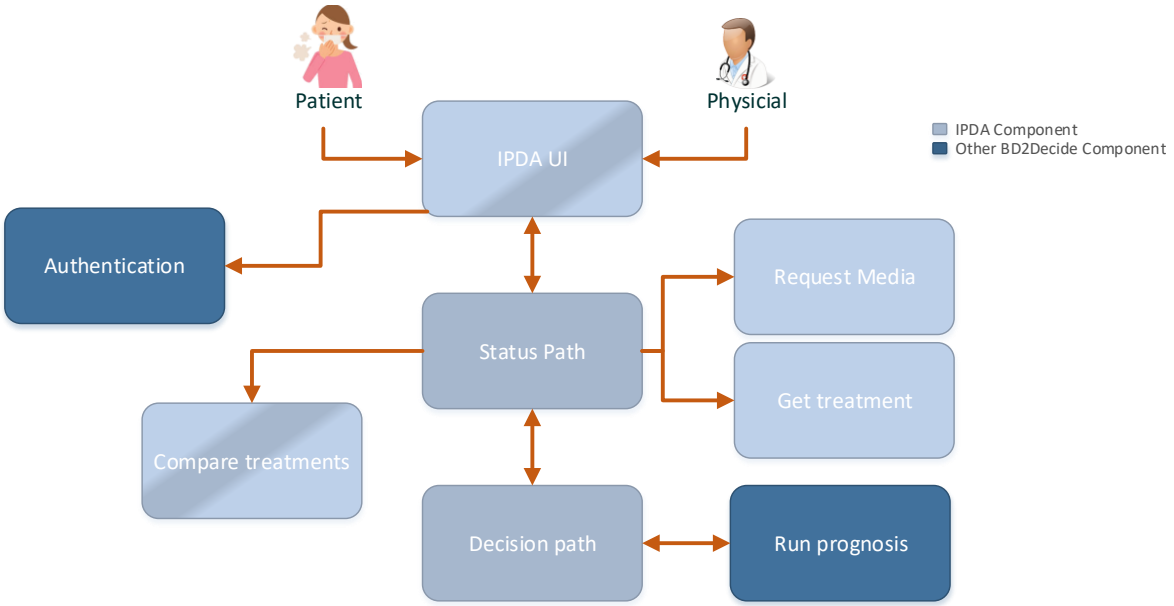


Figure 8 - IPDA architecture

The main modules of the IPDA are the:

- **IPDA UI**, which is the user interface that both patients and clinicians will use. This UI includes all the interactions with IPDA and has to be simple and understandable by users with basic computer skills.
- **Content Database,** which includes all the content, multimedia and textual, that is displayed in the IPDA UI. Multimedia content can include videos, animations or sound files that provide information about the available treatment methods.
- **IPDA Orchestrator,** which is the module responsible for the communication of the IPDA UI with the Content Database as well as with other external modules which offer additional information to the users. The Orchestrator is also responsible to track the current status of the IPDA workflow and invoke the modules required at the next state.

Additional modules that provide part of the IPDA functionality but are not building blocks of it include the authentication and the prediction modules. The first one can be used to allow authenticated patients pass their personal details to the IPDA so as to receive personalized information. The latter one provides statistical information and prognosis of the results of a treatment based on the prediction analysis made by the respective module.

2.4.4 Imaging models architecture

This section covers the Fraunhofer image analysis software, the radiomics feature extractors of POLIMI and MAASTRO, and the POLIMI phenotipization tool.

The Fraunhofer image analysis tool, allows the clinical partners to create segmentations of the tumor and the lymph nodes of the patients and extract relevant imaging features. The

image analysis software will retrieve the medical image data from a local clinical data repository. This is a huge benefit for performance and the clinical image data does not have to leave the hospital to be processed. All information processed by the software will be automatically stored on the clinical data repository and is thereby, available from any other computer running the image analysis software. In addition, once the big data infrastructure is available, the extracted features and segmentations can be transferred and stored there.

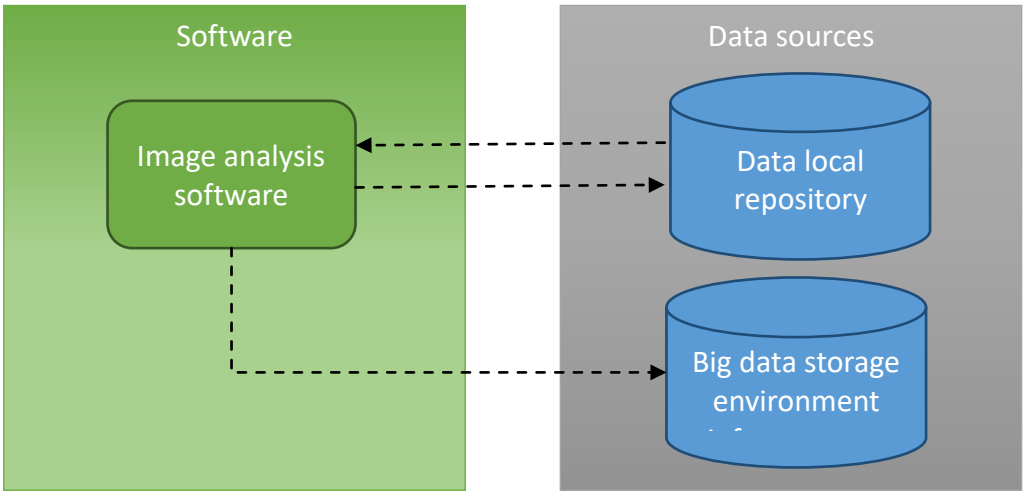


Figure 9 - Segmentation imaging software components

POLIMI will provide two tools: the Feature Extractor (FE) and the Phenotipization Tool (PT). FE will extract features from images provided by the BD system on ROIs identified by Fraunhofer software. After all retrospective patients will be analyzed by the FE, the features database will be the input for the PT along with clinical outcome and clinical characteristics provided by the BD system. The output of the PT will be a signature, i.e., a subset of features, selected as the best features able to differentiate the different patients' outcome.

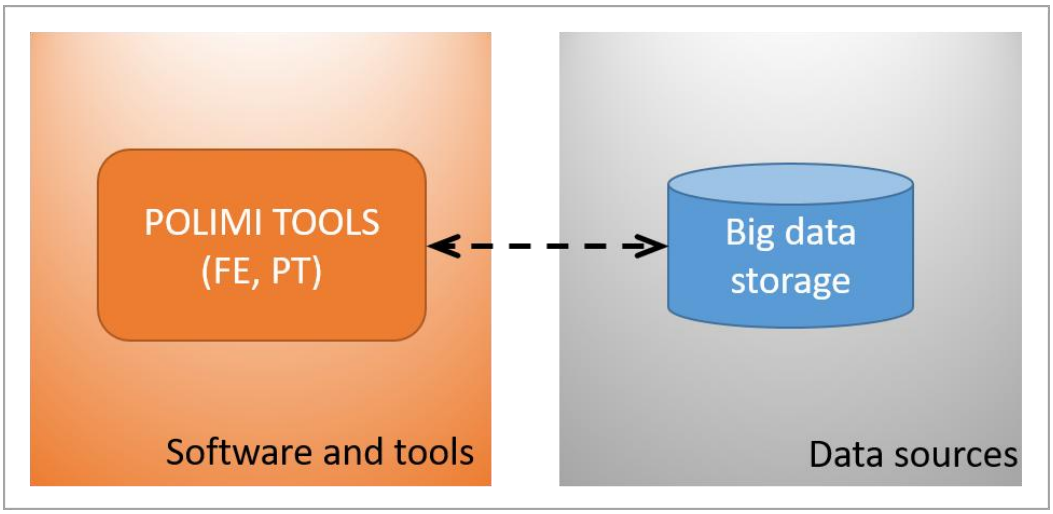


Figure 10 - POLIMI tools (FE and PT) and BD big data system relation

MAASTRO will receive all CT image data and their corresponding tumor and lymph node segmentations. The will use their CT radiomics feature extractor to extract the radiomic features for all clinical centers. The extracted features will then be transferred to the big data infrastructure. The output if the software will be the radiomic feature values in a comma-separated text file (Figure 11).

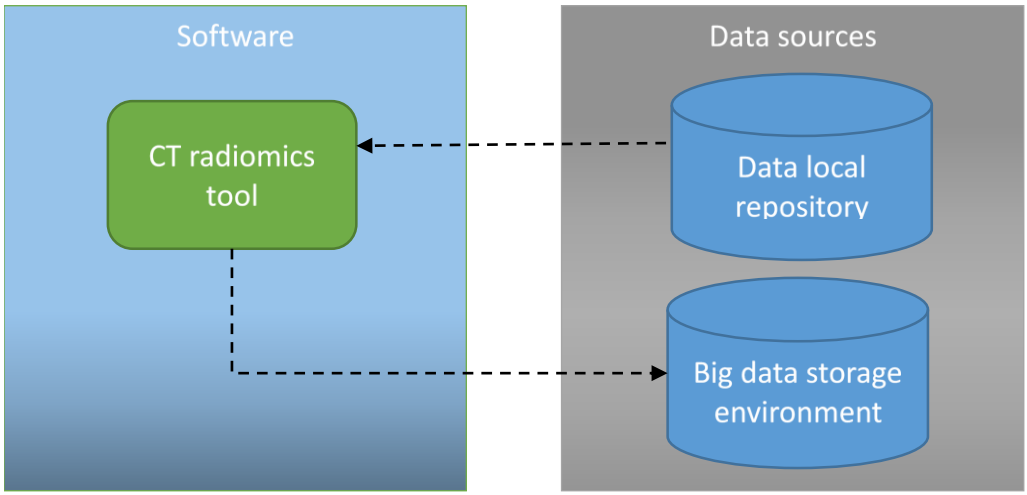


Figure 11 - MAASTRO imaging software components

2.4.5 Big Data Infrastructure

All-In-Image (AII) delivers the database management system and the knowledge discovery system for the semantic search and big data analytics for BD2decide. The database management system is based on compressed datafiles are based on AII 7X-COLORS protocol (provisional patent application submitted), which is used as the storage model of the project handling heterogeneous big and complex data coming from multiple sources. In All-In-Image database management the system dataset files are persistent PNG electronic images called CLAFs. Each CLAF contains various image tags (GEOTAGS), which are used to automate contents location for fast retrieval and searching. Each file shard within the database is a self-contained encapsulated dataset containing compressed metadata and data, which can be easily copied and moved between processing elements and storage tiers across the network. The All-In-Image data management system dynamically handles large catalogue of attributes and metadata members for enabling the semantic queries and provides the information retrieval for the BD2decide project.

The All-In-Image database management system is shown in Figure 12 and has the following components:

- **Data Import services**. This component allows to upload every file format of type CSV, JSON, XML, n-Triples, OWL-XML which is coming from multiple sources in BD2Decide project. The main importing services recognize the uploaded file format. It then catalogue each data member, format the data to the propriety storage format called CLAF, and save this in the database file system storage for retrieval component.

- **Rest Authentication Service.** One of the primary methods for communicating between components in BD2Decide is the REST communication service API. The client using the BDI will change credentials for an *authentication token* with the All-In-Image database management system, and in subsequent requests will send this token. In a production deployment, a secure HTTP must be used to protect the password in transit. The authentication token provided by the All-In-Image database management system encapsulates the user access privileges, session control and session expiration.
- **SQL/Big Table Query Processor.** This is query processor allows to extract the big data table (CSV like output) based on query parameters and which is being used for statistical analysis, visualization inputs, and machine learning purposes.
- **SARQL Query Processor.** This is the semantic query processor for retrieving and manipulating data which are stored in RDF format type in the CLAF data file, and is being used for the knowledge discovery based on the BD2Decide Ontology Galaxy to serve BD2decide analytics goals.
- **Low Level API.** This API can be used to access the database without requiring bulk of the data transfer over the network.

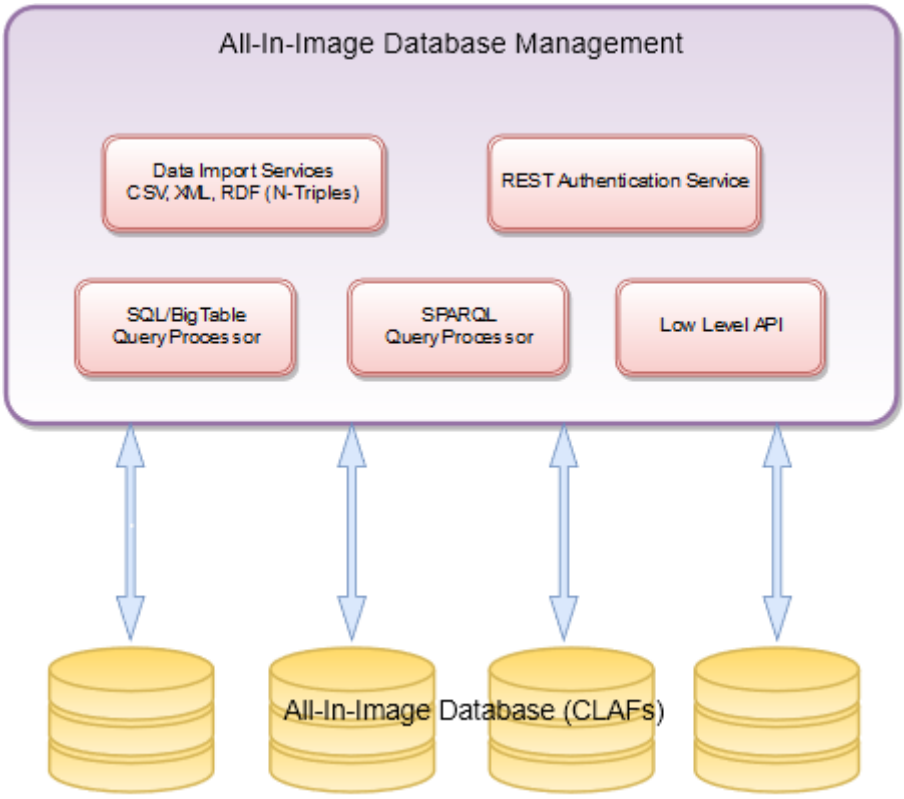


Figure 12 - All-In-Image database management system

The All-In-Image data management system includes two main services

- All-In-Image import services, which efficiently handle any data formats of heterogeneous big and complex data coming from multiple sources such as

BD2decide and is described in details in section 3.5. The All-In-Image data management system dynamically handles large catalogue of attributes and metadata members for enabling the semantic queries and provides the information retrieval support to the following outputs required in the BD2decide project:

- The All-In-Image information retrieval services which provides:
    a. The big data table (CSV like output) data retrieval, which is being used for research, statistical analysis, visualization inputs, and machine learning purposes.
    b. The SPARQL implementation, which is the semantic query language for retrieving and manipulating data stored in RDF format, and is being used for the knowledge discovery based on the BD2Decide Ontology Galaxy to serve BD2decide analytics goals.

2.4.5.1 Data semantic and data linking

The All-In-Image semantic database creates an environment that combines the medical ontology developed in the project with the rest of data collection to support the development of diagnostic assertions for use in screening for the disease and other clinical conditions. This functionality is built on the following steps:

- An ontology is designed to represent the class hierarchies for essential medical concepts (diseases/conditions, therapies, clinical observations, outcomes, and procedures) and to capture the relationships between these medical concepts.
- We develop tools for extracting collections of concepts from other ontologies that are relevant to a specific clinical question and are related to research. For example, retrieve data from recognized classes of medical data and standard medical terminologies such as the International Classification of Diseases (ICD-10), LOINC and SNOMED.
- We develop tools for data retrieval using free-text clinical documents represented by these concepts from internet resources such as MEDLINE. For this purpose, we demonstrate the use of natural language processing tools and extract statistical data and information using PDFMiner, PDFQuery, python slate and others.

The All-In-Image semantic SPARQL query processor makes it easier to develop reusable techniques for querying, exploring, and using data. The data semantics is based on entity-relationships modelling between keywords in the project, and linking the data to other resources is often just a matter of aliasing or connecting a few keywords.

2.4.5.2 Knowledge Management System

The Knowledge Management System (KMS) provides a platform to convert data into information with semantic value and extract knowledge from it. The KMS stores and retrieves knowledge, improves collaboration between modules in the system, locates

knowledge sources, captures and uses knowledge, and enhances the knowledge management process.

In the architecture of the BD2D system a fundamental component is the KMS. This module is the main manager of all system information.

The KMS structure the data and use the information in order to perform tasks and make decisions based on the requirements, knowledge and experiences acquired in the cancer domain.

The KMS consists in two main components (Figure 13):

1. *Ontology Galaxy.* Through the *Ontology Galaxy* all the knowledge related with H&NC and used inside the BD2Decide system is modelled. The knowledge are modelled through the definition of terms (concepts and classes) inside the *Ontology Galaxy* and also the properties that relate these concepts. All data used in the BD2Decide system are mapped and structured in the concepts and properties defined in the ontology. Also a set of basic rules are defined as a part of the *Ontology Galaxy*. This rules can be interpreted as a first-order rules and are based on the relationships that exist between the data to provide them with semantic value.

2. *Reasoning.* This component contains a set of rules that are grouped in different levels. The aim of *Reasoning* is to analyze facts of knowledge base and use the rules to solve conflicts, deduce and infer possible solutions and store strategies for the future. Rules in basic level are based in the data extracted directly from the data sources and the logic associated to this data is very basic. In the top level the data used are the outcomes of the basic rules, correlated with other external data, and the logic used is the result of the enriched knowledge through specific conditions. This component also includes a system learning module, which allows the updating of the knowledge base in terms of the solutions obtained and subsequent developments of the facts, and also updates the schemas of interpretation of the data model.

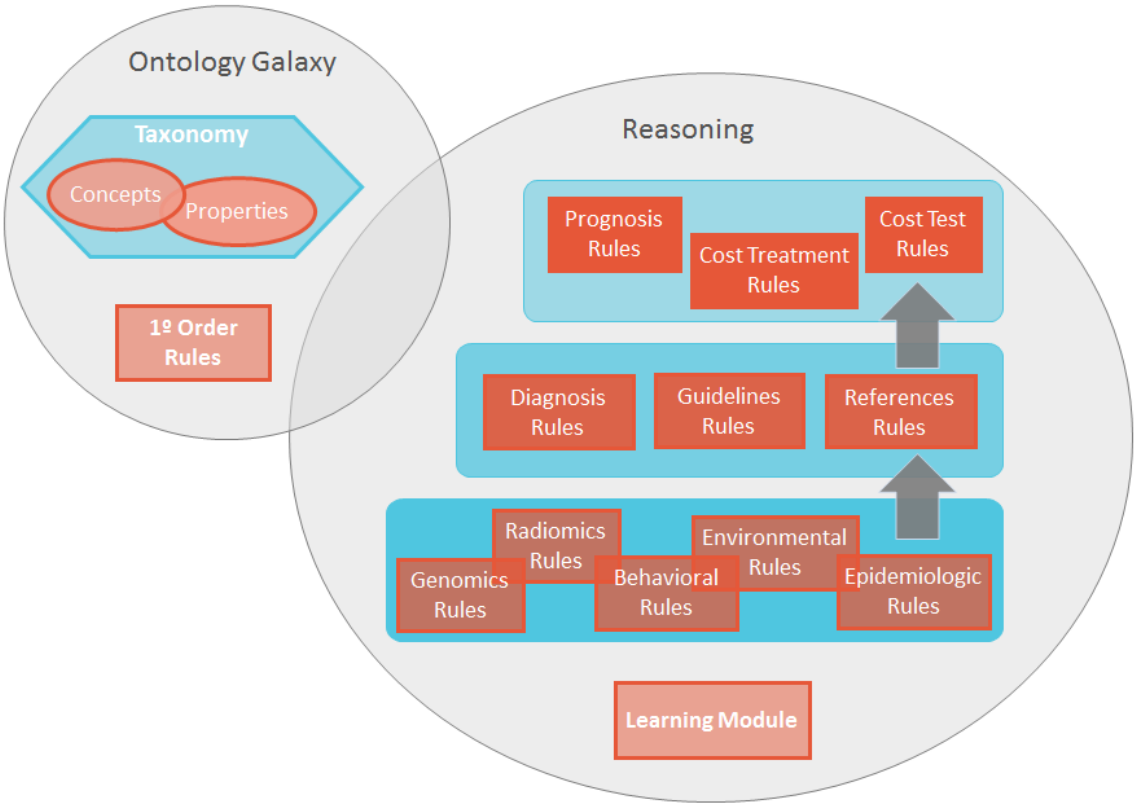


Figure 13 - Knowledge Management System components

The Figure 14 shows the interaction between the components of KMS and other modules of BD2Decide architecture. The concepts defined in the Ontology Galaxy are used by the RDF Parser to translate data exchanged between tools (i.e. CDSS, VAT and IPDA apps) and APIs developed for querying and interchange data with the BDI. APIs details are available in the Deliverable 6.1.

The RDF Parser also uses the ontology concepts to guarantee the consistency of the data received by the APIs and ensures the interoperability between the tools developed, since all the applications use the same data model defined in the ontology in order to guarantee the data consistency.

The KMS makes use of the knowledge base that consist in a set of information based on guidelines, clinical practices and information from cancer researches for generating, using and updating the decision rules defined in the Reasoning component.

The Decision Rules defined in the Reasoning component are handled as inputs to train and test the machine learning algorithms implemented in BDA and employed for discovering relevant information related with the improvement treatment in the cancer domain.

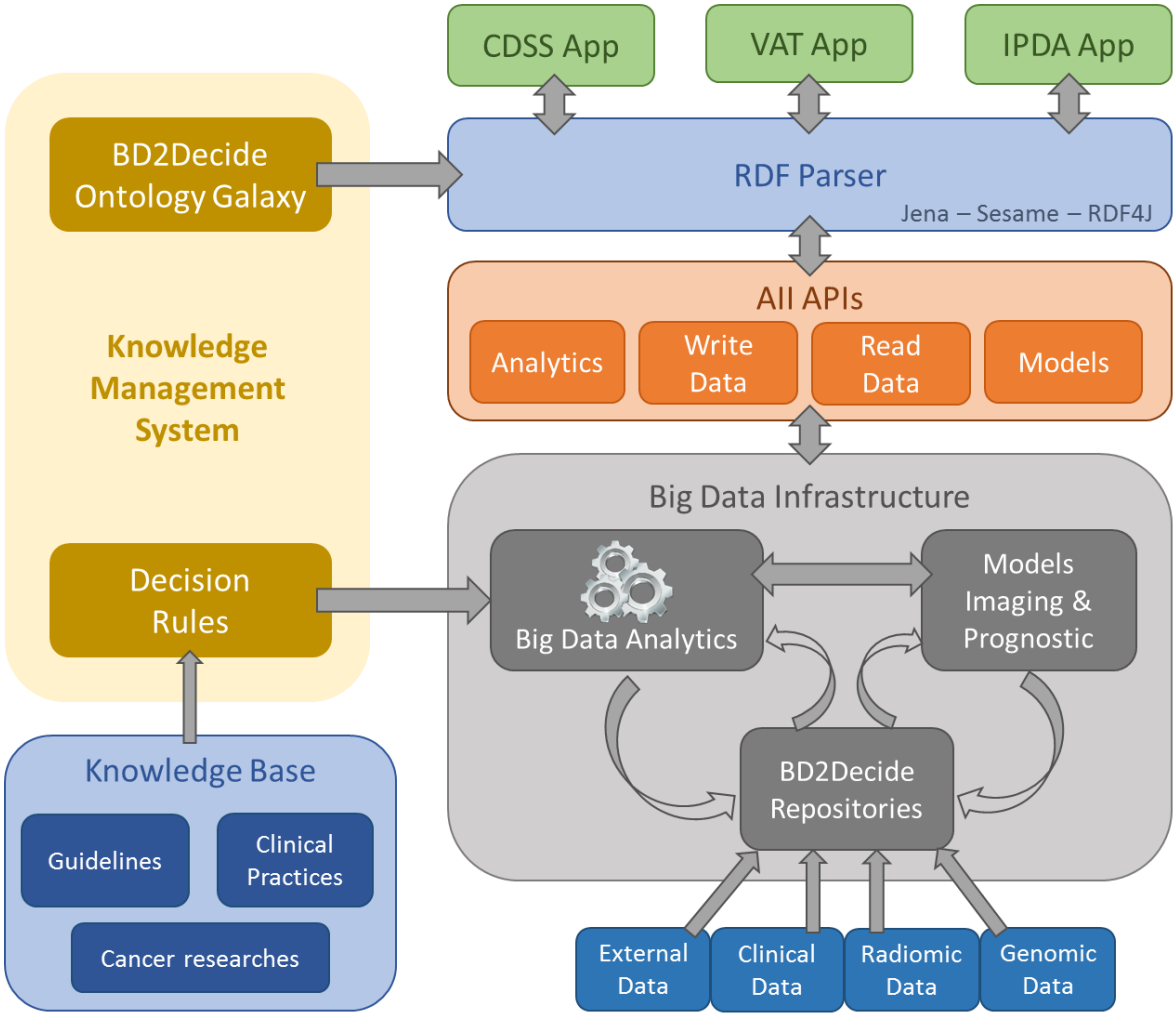


Figure 14 - Knowledge Management System interaction

The Figure 15 represents the technologies used by the KMS. Technologies have been organized by levels in order to details the different languages used to describe and querying the semantic data, and also to define and express the inference rules.

In the lower levels we have located the UTF-8 character encoding utilized to define the character set used in the Ontology Galaxy concepts definitions and for decision rules. Also in the definition of the Ontology Galaxy have been used the URI (Uniform Resource Identifier) format for the identifiers of class and properties. This format allows identify each concept with a unique string of characters.

The OWL2 (Web Ontology Language) language is used for the general definition of the Ontology Galaxy and the RDF (Resource Description Framework) format have been used to define the taxonomy and data interchange inside the ontology, since this format is used for conceptual description and information modelling. The RDF format use a variety of syntax notations and data serialization formats but we have used an XML/JSON syntax as

communication protocol between KMS and others components of the BD2Decide architecture, such as Big Data Analytics module and tools inside User Interfaces Layer.

The SPARQL (Protocol and RDF Query Language) language is used for querying, retrieve and manipulate data stored in RDF format. This language allows create queries using aggregation, negation, constrains and subqueries.

The Semantic Web Rule Language (SWRL) is used to express the decision rules defined in the KMS, this language allows define rules as well as logic between the concepts inside the ontology; and the Rule Interchange Format (RIF) is used for facilitating the develop and exchange of rule sets among the different modules of the BD2Decide architecture.

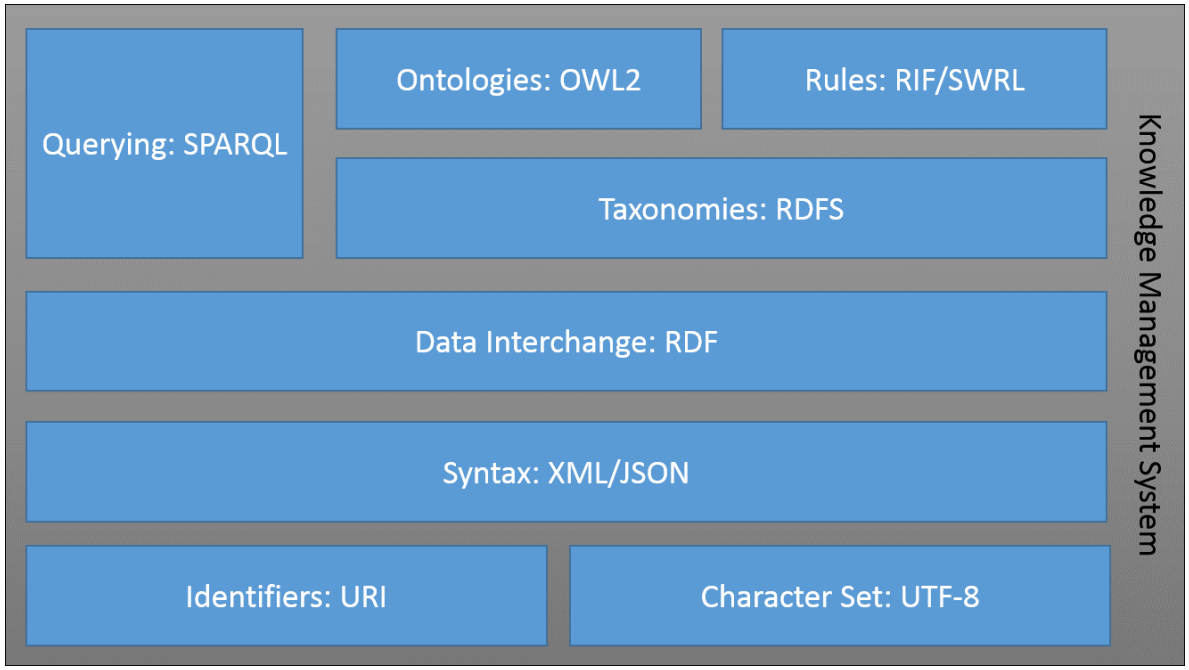


Figure 15 - Knowledge Management System technologies

2.4.5.2.1 Ontology Galaxy

This section includes a brief description of the main components and characteristics of the BD2Decide Ontology Galaxy. This information will be detailed in the deliverable D6.4 that is about of the Knowledge Management System features documentation.

The BD2Decide system will deal with information of different nature, source and formats and these data will be stored in several repositories and will be used for different tools. For this reason it would be useful define and create an Ontology Galaxy that help to handling the integration, normalization and linked all data in the system.

Ontology Galaxy provides a formal description of shared data, so that application programs and databases are able to interoperate without sharing data structures.

The BD2Decide Ontology Galaxy will defined and created based in the ontology defined in the European project Neomark[2] that it's focused in the prediction of cancer reoccurrence. This ontology refers to the terms of the oral cancer and is also used in the Oramod project[3].

Others pre-existing ontologies are also merge in the definition of the Ontology Galaxy in order to reuse concepts in the biomedical, human diseases, environments and literature domains (OBI[4], DOID[5], BIBO[6] and CTO[7] Ontologies).

Four main sections are defined in the BD2Decide Ontology Galaxy:

- *Virtual Patient:* This class models all information related to the patient, includes demographic, clinical, pathology, imaging and genomic data of patients, risk factors of developing H&NC and information of the treatment, follow-up and quality of life.
- *Head and Neck Cancer:* This class models the specific data related to the tumors, lymph nodes, data extracted from medical images, possible kinds of malignant headplasm and the different existing treatments.
- *Context:* This class models the contextual information of patient that will help diagnose and prognosticate head and neck cancer diseases.
- *References:* This class models the information related with references and literature including existing scientific papers, clinical guidelines, and articles.

The class *Virtual Patient* was defined taking into account the data defined in the eCRF [See e-CRF scheme (Annex I to D2.1)]. During the mapping process, the sections and data defined in the eCRF document have been included in the ontology.

A set of rules are defined to relate the different concepts included in the ontology. These relationships are essential for ontological purposes in order to assure semantic value to data, and more accurately model the domain of knowledge.

The ontology editor Protégé[8] are used to implement the Ontology Galaxy and the Description Logic (DL) and its machine triable implementation, the Ontology Web Language (OWL) are the formal languages used.

[2] http://cordis.europa.eu/project/rcn/86607_en.html

[3] http://www.oramod.eu/

[4] http://obi-ontology.org/page/Main_Page

[5] http://disease-ontology.org/

[6] http://bibliontology.com/

[7] https://bioportal.bioontology.org/ontologies/CTO

[8] http://protege.stanford.edu/

2.4.6 Statistical models architecture

The BD2Decide prognostic modeling tool has the purpose of calculating a prediction of a pre-defined outcome (e.g., survival probability or hazard) for a new patient given input data consisting of clinical, histopathological, genomics and radiomics data of this patient. In addition to a prediction (i.e., one scalar number per patient), the modeling tool provides upper and lower confidence bounds (two scalar numbers per patient).

In the diagram shown in Figure 16 the elements of the BD2Decide system involved in the Statistical Models are presented. The continuous lines represent the direct access to the models (inputs and outputs) and the discontinuous arrows represent the indirect relation with those elements who take part of the process but not directly, that is, that have some dependencies with the statistical models.

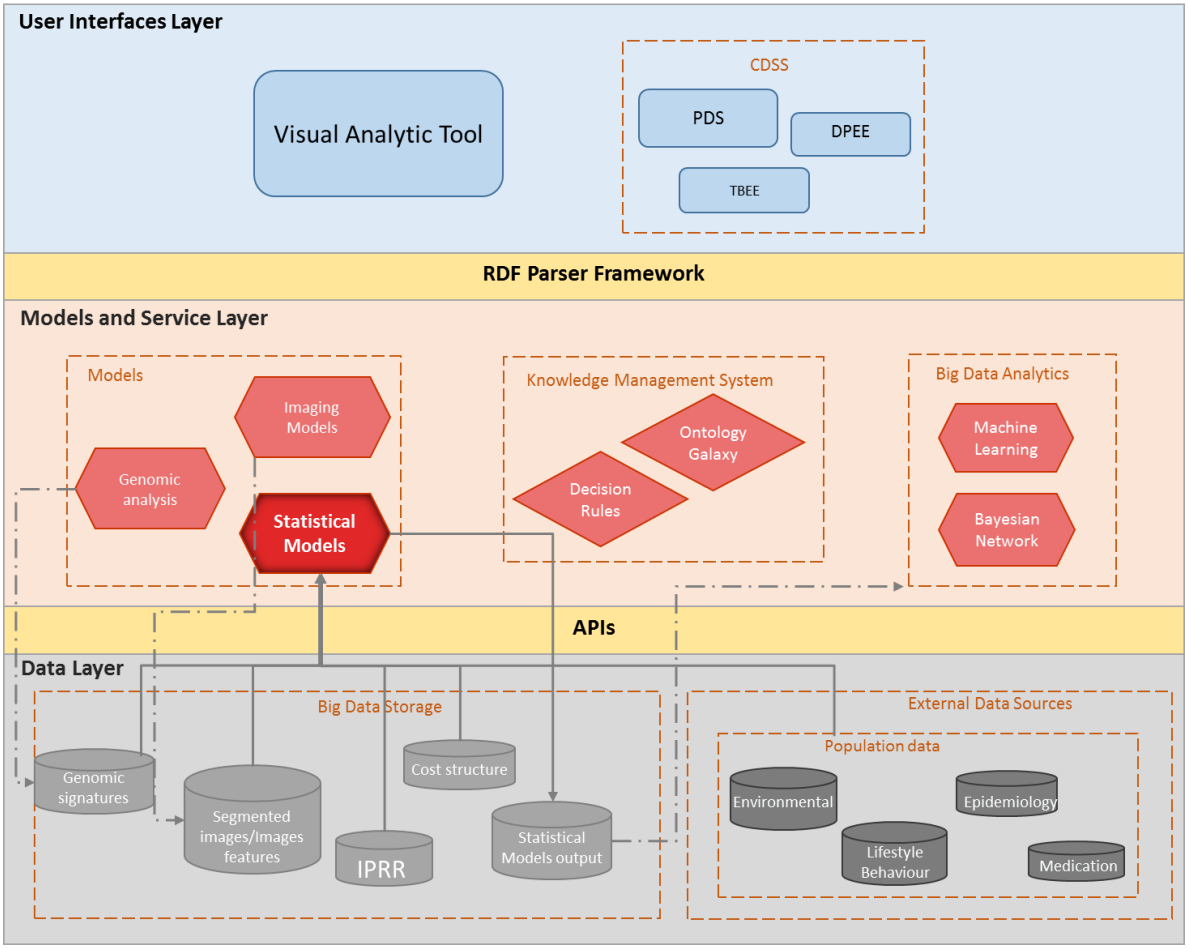


Figure 16 - Statistical models architecture

As shown in the architecture diagram, the statistical model is feed by clinical data (IPRR represents the e-CRF content, but only a sub-set of the complete data will be used by the final models), genomic data (Genomic signatures repository), radiomic features (this features are also included in the Segmented images/Images features) and population data.

Additionally, for the 'Cost-utility analysis tool' the models need also the feed of the cost structure database. Genomic signatures are provided by Genomic analysis of the raw genomic data, and radiomic features are result of the imaging models.

The outputs of the statistical models are stored in the 'Statistical Models output' and then this data will be used in the Big Data Infrastructure for further analysis (BDA, if required) and for the tools: VAT and CDSS in order to provide these predictions to the clinicians.

VU/VUmc workpackages 4.3 (model updating tool) and 4.4 (cost analysis tool) will be developed in the future as specified in the BD2Decide project time plan. For this reason, detailed information on the requirements are currently not available. However, we expect both tools will have the same basic input requirements as the prognosis tool (4.1 and 4.2), i.e. an R environment to run the model functions on and data in tab delimited format as specified above.

Specifically, model updating (4.3) will require the same **input data** (clinical, genomic, radiomic) as the model function in the same format (tab delimited .txt). It will **output** a vector of updated model coefficient parameters in a tab delimited .txt or an .R data file. This updated parameter vector can be used by the modeling tool (developed in 4.1 and 4.2) to make prognoses based on updated parameters. For this purpose we will design the modeling tool (4.1 and 4.2) flexible enough to read new parameter vectors that are output from the model updating tool (4.3).

Furthermore, the cost utility analysis tool (4.4) will require as **input data** again the clinical, genomic, radiomics and/or histopathological predictors of a new patient (typically only predictors from a subset of these modalities are needed, where cost-utility will be assessed for the other modalities separately). Furthermore it requires a set of cost data (in Euros) for data collection in a given center (e.g. a cost for obtaining genomic data, a cost for obtaining the patient's imaging data and costs for histopathological examination). Cost structures can vary across countries and centers. Therefore each center wishing to use the cost analysis tool needs to submit its own cost data. Subsequently, the cost analysis tool evaluates whether for a given patient collecting additional data is beneficial given the cost. The **output** is an estimate that will tell clinicians the gain in accuracy of outcome prediction by collection additional data (a scalar or a vector in .txt tab delimited format). The output should be displayed to the clinician and stored with the patient's records.

3. BD2DECIDE TECHNOLOGIES

This section intends clarify to all the technical specifications involved in the whole BD2Decide. First of all, all BD2Decide components will have the feature of being modular. Modular design, or 'modularity in design', is a design approach that subdivides a system into smaller parts called modules or skids, that can be independently created and then used in different systems. To get this feature off the ground a modular programming has to be used. Modular programming is a software design technique that emphasizes separating the functionality of a program into independent, interchangeable modules, such that each contains everything necessary to execute only one aspect of the desired functionality.

As mentioned before one of the components that involves all BD2Decide system is the Ontology Galaxy. The ontology framework is being developed through Protegé. Web Ontology Language (OWL) is the language used to represent ontologies explicitly, it is structured in layers that differ in complexity and can be customized. The ontology will provide the data normalization that allow the interoperability of the whole set of tools implemented and integrated within the BD2Decide system. The interoperability of the applications and the usage of the data will be carried out through the APIs that are being developed. These APIs are introduced in this document in the section 3.4, but in D6.1 these are explain deeper.

Before introducing each component of the BD2Decide system specifications, in this paragraph a brief description of the interoperability taken into account in the whole system is described. These features make each interface work with other products or systems without restrictions.

The collection of data in the clinical centers is based on widely adopted clinical standards such as HL7, ISO/CEN 13606 and openEHR, adopting the IHE Cross-enterprise Document Sharing (XDS).

The user authentication in the tools follows the OpenID standard. This standard facilitates for single-sign-on by providing mechanisms for both authorization and authentication processes. Both request and response message formats are defined to facilitate the transmission of necessary credentials within a Web service activity. It is decentralized standard, meaning that it is not controlled by any website or service provider. In D5.1 more details about this standard can be found.

To allow secure authorization in a simple and standard method from the web application developed during BD2Decide project, Open Authorization (OAuth) open protocol will be used. It allows users to share their private resources stored on one site with another site without having to hand out their credentials. This protocol is also detailed in D5.1.

Having said that, the specifications of each BD2Decide component is detailed in the following subsections.

3.1 User Interfaces Look and feel

The technology solutions employed for the application's user interaction are strongly coupled with the available end user devices and the purpose of the envisaged interaction. Currently, a plethora of available devices are being used, to enable users interact with software and systems from intuitive User Interfaces (UIs). Such end user machines range from Web servers and desktop PCs to mobile devices and tablets, bringing to different capabilities, which should be taken into account from a UI definition and development's point of view. Thus, the choose of the UI design and development constitute a key point on the adoption of one over another programming language for the UI development.

Another criterion that should be taken into consideration is the goal of the end user interaction through the software application. Whether the user has to consume an online service or has to experience with local software to manipulate digital data is of major importance as well.

Regarding technologies, depending on whether the user interface is a desktop or a web-based application, a variety of programming languages can be used. The main criteria have to do with the fact that the technologies should be easy, user friendly and be supported by an active community, so that maintenance capabilities are maximized.

In BD2Decide, the user requirements dictate the creation of an interface that will be mainly accessible through a desktop pc and which should allow easy access to a great amount of clinical information related to patients. The existence of many different components which provide separate pieces of information, using different backend technologies, lead to the need of a modular design, as described in the previous paragraph. Therefore, the interface should be totally separated from business logic and only focus on the visual representation of the data and user interaction. The general architecture that allows this setup involves a web application which communicates with the backend via a set of RESTful web services.

The basic technologies for the BD2Decide web applications will be HTML5, CSS and JavaScript. The latter will be used to perform advanced user interactions and for the visualization of various data. These functionalities will be realized through the use of JavaScript libraries or frameworks, such as Angular or React for the user interactions and Google Charts or High Charts for the data visualizations.

Moreover, a common visual identity will be followed by all frontend components. Basic colors, form styles and interaction elements (e.g. buttons) will have a common look and feel, so as to provide users with a unified experience when using the BD2Decide tools. Such solution does not cause that the existing interface guidelines, for some of the

components, will be disregarded. For example the implementation of the EQ-5D[9] questionnaire will comply with the rules published by EuroQol.

Details of the specific technologies for the UI of each component is available at the next paragraphs. The Look & Feel for each tool will be presented in D5.3.

3.2 Clinical DSS tool suite technology

The CDSS incorporates a set of diverse tools in order to be able to satisfy the needs of clinicians. Some of them require specialized technologies or include already available modules which are developed in specific technical setups. This being said, the following list presents the technologies that will be used by the identified modules of the CDSS. It must be noted, that these descriptions reflect the current status of development and are subject to changes as the project progresses and newer user requirements emerge.

- **Dashboard module:** This module will follow the Single Page Application (SPA) approach so as to provide a quickly adapting, data-driven and easy-to-use interface to the clinicians. A JavaScript framework like Angular.js or React.js will form the basis for such an interface. jQuery and Ajax will be used to communicate with the databases via the services that will be offered in a RESTful style.
- **The e-CRF management module:** The retrospective data gathered have been inserted to the system using OpenClinica as a commonly used and known tool by clinicians. OpenClinica runs as a java application using a Tomcat server and PostgreSQL as the database server. The e-CRF management module will therefore follow the same technical setup and will be developed as a java application, performing all necessary functions that will connect OpenClinica (used for collecting H&NC patients retrospective data) with PDS (used for collecting H&NC patients prospective data) and CDSS.
- **The tumor localization tool:** This is a graphical tool allowing for a user to draw on an existing image. JavaScript will be the main technology driving this tool, with accompanying services developed as a java application to support backend communication needs.
- **The DPEE UI:** This UI will be built based on STL format for the representation of the models and a STL rendering framework like e.g. Three.js which is compatible with both HTML5 canvas and WebGL.
- **Prognostic Prediction Visualizations**: Visualizations used in this module have to be interactive and integrated in the CDSS. Highcharts.js, a well-known charting library based on JavaScript will be used for this purpose. Highcharts are also tested and easily integrated with the aforementioned JavaScript frameworks that will be used for the Dashboard module.

[9] http://www.euroqol.org/about-eq-5d.html

- **Tumor Board Organizer:** The current plan for this module is to use an existing framework that offers the functionality required, namely online collaboration capabilities, online audio and video conferences, etc. Apache OpenMeetings is such a tool, which is an open source java application that can be deployed in PostgreSQL.
- **The QoL assessment tool:** In order to implement the questionnaire functionality of this tool, the LimeSurvey survey tool will be used. LimeSurvey is a PHP based application that can handle the creation and management of surveys and their results. The QoL questionnaires have been created as LimeSurvey surveys and the results will be saved on the application's database which is built on PostgreSQL. A custom middleware will calculate the QoL scores and save them to the PDS so they can be available for the CDSS when viewing a patient.

Summarizing the above, the main technologies used in the CDSS are:

- Programming: Java
- Frontend: JavaScript, React.js, Highcharts
- Backend: Tomcat application server, Apache web server
- Database: PostgreSQL
- Communication: RESTful services

3.3 Visual Analytics tool technology

This chapter aims to cover the technical specifications of the researcher tool. In the Figure 17 a layer diagram of this tool is presented. In this, different levels of layers appears. As shown, the user (researcher in that case) has access only to the user interface, this presentation layer alludes to the interface elements (UI components) such as input controls (checkboxes, radio buttons, list boxes, etc.), navigation and informational components (slider, tags, tooltips, icons, notifications, etc.) and containers; but also the components needed to synchronize and orchestrate user interactions maintaining thus the process flow.

Related with the front-end layer, the application framework is the responsible for providing a fundamental structure to support the development of the application, in other words, it acts as the skeletal support to build the researcher tool.

On the other hand, the application logic layer contains the rules that determine how data can be treated and the interaction and workflows between the diverse tasks of the system. Furthermore, the access to the data will be through REST APIs that are in development within the BD2Decide frame.

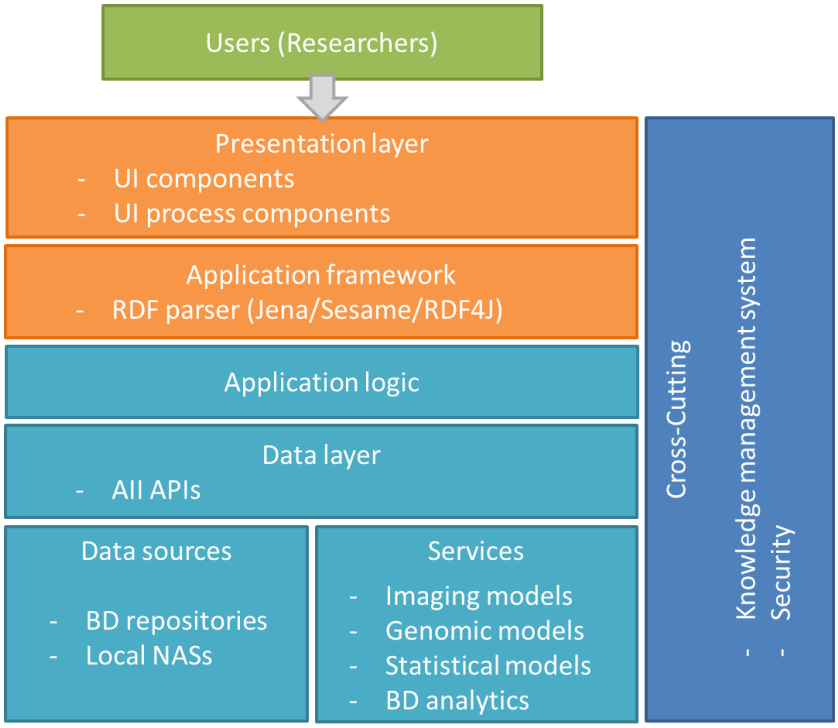


Figure 17 - Visual analytics tool layer diagram

These APIs are going to provide access to data located in the BD repositories and also to manage the models involved and developed in the project: imaging, genomics, statistical models and also the techniques developed within the BDI, BD analytics.

Affecting all layers the cross-cutting concern forms the basis for the development of system features. It includes the knowledge management system and the security aspects.

Once to this point, the specific technical requirements are defined in the Table 1 and in Figure 18.

Requirements	Formats/languages
Application type	Web
Web browser compatibility	IE, Firefox, Chrome, Safari and Opera
Language of the user interface	English
Front-end technologies and frameworks	HTML5 and CSS. Javascript (JS) integrated in the HTML5
Back-end technology	PHP/Python
Data access	All-in-image APIs

Table 1 - VAT technical requirements

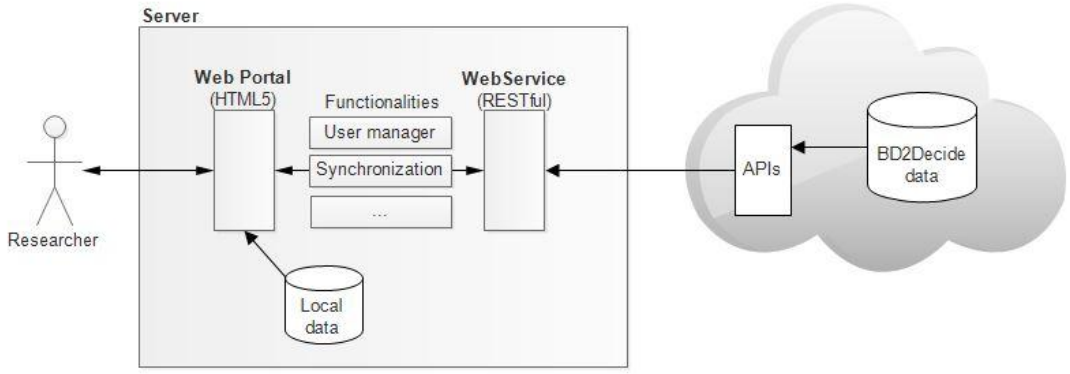


Figure 18 - VAT enterprise application diagram

For the visualization of the data it is proposed to use Google Charts. These tools are powerful, simple to use and free, allowing to connect the dynamic data in real time. One of the features allowed are to connect charts and controls into an interactive dashboard, and all the available charts are customized to create a visualization suite in accordance with the user requirements.

Development and deployment details

The tool will be developed in PHP with a PHP framework such as Yii. The server where the tool is going to be deployed will have the following requirements: Linux, Apache and MySQL and it will provide access by IP. In order to provide interoperability we are going to use web services and for the front-end technology HTML5 based on an existing template.

The VAT will need a local database for the internal logic and the web service will help to access to this database. The data will be available in real time thanks to the synchronization services.

For the deployment, the web application will be integrated in the hospital intranet.

3.4 Interactive Patient's co-Decision Aid technology

The IPDA is a web application which aims to provide a simple interface to present informative material of various formats. The technologies comprising the IPDA are:

- **IPDA UI**: An HTML-5 based webpage utilizing JavaScript (jQuery) for the manipulation of the various transitions between the IPDA's sections. Ajax calls are used for the communication with the backing web services.
- **Content Database**: A relational database (MySQL or PostgreSQL) storing all the necessary material to support the IPDA UI. Multimedia content can be stored in the database or at external sources (e.g. YouTube for videos). In the latter case, pointers to the content's location will be stored in the database.

- **IPDA Orchestrator**: The orchestrator will provide a set of RESTful web services to enable the communication of the IPDA UI with the Content Database. The orchestrator is a java application requiring an application server such as e.g. Tomcat to run.

Moreover, a prototype of the IPDA is created using Storyline Articulate (https://www.articulate.com/), which is a software to create interactive content. The tool converts the code into HTML5 or flash code, which can be integrated in the BD2Decide system. To create the multimedia content, 3D animations, graphics, and videos software such as Adobe, Maya and Camtasia will be used.

3.5 Big Data Infrastructure technology

All-In-Image database management system has a generic importing services, which can handle efficiently any kind of data type, such as XML, JSON, CSV, DICOM, RDF n-Triples, and is highly robust in storing unstructured large heterogeneous complex data from multiple sources such in BD2decide project. The collection of all data coming from multiples sources into All-In-Image central database management is summarized in Table 2.

There are two main services in our BDI: information retrieval/querying service and data files uploading service.

- **Uploading Service**. For data uploading into the BDI and storing it in the CLAFS storage, we have a special directory called "arrivals" and subdirectories for each data provider. The incoming directories are located in the server in which All-In-Image software is installed. Every data provider or component tool that will require to transfer its data to its own subdirectory be able to constantly perform this by secure data transfer. Once the data is uploaded, the All-In-Image infrastructure will automatically monitor any new incoming data, parse the data, catalogue its data members, and store it in All-In-Image CLAF format along with the date and time stamp.
- **Information Retrieval Services**. The supported output required in the BD2decide project:
    - The big data table (CSV like output) data retrieval which is being used for statistical analysis, visualization inputs, and machine learning purposes.
    - The SPARQL implementation, which is the semantic query processor for retrieving and manipulating data stored in RDF format, and is being used for the knowledge discovery based on the BD2Decide Ontology Galaxy to serve BD2decide analytics goals.

This section describes the semantic data inputs for the big data infrastructure used for querying the data with SPARQL query processor.

- **Patient Personal Data eCRF (OpenClinica).** The OpenClinica tool is selected to design a clinical patient study. The patients' health records, also known as the e-CRF records are saved in the OpenClinica tool, and then exported to an XML file. This file is imported to All-In-Image data management system, where the uploading format type is XML. It includes all the fields comprising the patient health records, and is stored in the CLAFs storage for the semantic queries usage based on the metadata mapping available in the BD2decide Ontology. The e-CRF is divided to various sections of the patients' information such as HNC stage, clinical data, risk factors, and other characteristics which are developed and collected from the treatment, scans, and information about chemotherapy and radiotherapy. [See e-CRF scheme (Annex I to D2.1)]. Based on the BD2Decide Ontology Galaxy, All-In-Image transforms the attributes and fields from the e-CRF patients' information to a collection of RDF triples, where in the main expressed RDF triple the SUBJECT URI is a PERSON (patient) defined by the project anonymous identity, and all the clinical data in every section are expressed as collection of RDF triples related to a patient. Each clinical activity is also expressed by a PREDICATE URI and a timestamp as its OBJECT value. Examples of clinical activities are: the diagnosis of the disease, and post events such as surgery, chemotherapy, radiotherapy, or other activities and their clinical results.
- **Fraunhofer imaging analysis tool.** This tool outputs the feature extraction and the segmentation data in JSON format. All-In-Image database import services upload this hierarchical structured data and relate these clinical data features to the patient anonymous id for both the big table (CSV) output representation and the SPARQL query processor.
- **POLIMI/MAASTRO radiomics feature extraction tool.** A similar process is available for POLIMI/MAASTRO radiomics feature matrices, where the incoming data elements are in CSV format.
- **Population data (INT).** The data arrives in CSV (excel) format.
- **Genomic Data (INT).** Since the genomic files resides in separate location with its own storage capacity, the sam-tools query scripts which will be developed at INT will be executed on this server, and their results set and indicators which are of interest the researcher will then uploaded with secure file transfer to All-In-Image database management system.
- **QoL.** The data arrives in CSV (excel) format.

The following table (Table 2) summarize various data inputs for the BDI system.

Semantic Data inputs for BDI	Partner responsible to provide the semantic data to AII	Input File Format	BDI Storage Format	BD2D related components
Images (Segmented) including METADATA + IMAGING FEATURES	FHF	JSON	CLAF/RDF	Radiomics Feature Extractors, VAT, CDSS
		DICOM	DICOM	
Radiomic Features (item 5a of PDS)	MAASTRO	CSV	CLAF/RDF	Prognostic models, VAT, CDSS
Radiomic Features (item 5b of PDS)	POLIMI	CSV	CLAF/RDF	Prognostic models, VAT, CDSS
Patient Personal Data (eCRF(OpenClinica) -->PDS) items 1,2,3,4,5,6,7,8,9,11,12	AOP, INT, USUS, MAASTRO, VU/VUMC	XML	CLAF/RDF	CDSS, VAT, IPDA, Prognostic models
Population Data	INT	CSV(EXCEL)	CLAF/RDF	CDSS, VAT, Prognostic models
Guidelines and References Data	AOP		CLAF/RDF	CDSS, VAT
Genomic Data (item 10a of PDS)	INT	Result set from sam-tools	CLAF/RDF	CDSS, VAT, IPDA, Prognostic models
QoL (item 13 of PDS)	ATC		CLAF/RDF	CDSS, VAT
Cost-utility analysis tool inputs	AOP, INT, USUS, MAASTRO		CLAF/RDF	CDSS, VAT

Table 2 - BDI inputs and storage formats

The system requirements for All-In-Image are described in Table 3.

Requirements	Formats/languages
Application type	REST service (customized to various application needs) Dashboard (data retrieval and a console for submitting queries during the research period) Daemon (watchdog for incoming data transfer used in the uploading service)
Cross-platform	All

Web browser compatibility	IE, Firefox, Chrome, Safari and Opera
Language of the user interface	English
Front-end technologies and frameworks	Flask framework as dashboard for browsing the data and running SPARQL
Back-end technology	Anaconda Python 3.4 and later
Data access	All-in-image APIs: REST (customized), SPARQL standard query endpoint

Table 3 - AII technical requirements

In order to maximize the data resource devoted to big data analysis, it was decided that the BDI will not serve as the storage host for the raw images and the genomic files, but only for the metadata, feature extraction from MAASTRO, POLIMI and Fraunhofer. Raw data inputs will be allocated on other local servers. It means that the raw images and genomic APIs access will be not handled within the BDI. The BDI will keep only trace of the image filenames, the paths and the hosts for each global patient id.

3.6 Imaging models technology

This chapter describes the technologies of the imaging tools. It will focus on the Fraunhofer image analysis tool, the POLIMI and MAASTRO radiomic feature extractors and the POLIMI phenotypization tool.

The Fraunhofer image analysis tool is the first tool to be used by clinicians in the image processing chain. The tool uses MR or CT DICOM image data as input. It will mount an on-site NAS system within the clinical environment, from which it will load the data. The clinician then creates a segmentation of the tumor and if present, segmentations of relevant lymph nodes. In addition the clinician determines some location-based information, for example, if the tumor has already infiltrated specific sites. The segmentations of the structures will be saved in DICOM-RT and nrrd file format and will be saved on the clinical NAS. The extracted features will be saved in JSON file format and stored on the NAS as well. This allows the radiomic feature extractor tools to directly access the image data and the generated segmentations. In addition the data will be send to the BDI using a REST API. From there the data can be accessed and used for additional steps, like for example the training of the statistical model.

Requirements	Formats/languages
Application type	Desktop executable
Cross-platform	Windows 7/8/10
Software requirements:	None
Hardware requirements	Full HD display, Core I3, I5, I7 CPU, 8GB RAM
Language of the user interface	English
Technologies and frameworks	ITK, VTK, QT, Boost

Table 4 - Fraunhofer image analysis tool technical requirements

Feature Extractor (FE) is a tool for calculation of volume characteristics (features) in digitized images that we specialized on magnetic resonance images (MRI). The FE tool will be applied on 12 images: 8 images directly obtained by the MRI scanner (namely, T1, T2 and 6 DWI images (b values = 0, 50, 100, 500, 750, 1000 s/mm^2) and 4 parametric images computed from the DWI, namely ADC, Fp, ADCslow and ADCfast. In particular, the ADC is computed as the slope of the linear regression of the logarithm of the DWI exponential signal decay on the b-values images. The calculation is performed pixel-wise by means of a least squares routine that evaluates ADC for each pixel. The computation of the ADC maps is performed in ITK 4.8. Similarly, Fp, ADCslow and ADCfast (the IVIM derived maps) are calculated from the bi-exponential fitting of the DWI images. The Levenberg-Marquardt solver optimization is implemented by means of Gnu scientific library (GSL), Basic Linear Algebra Subprogram (Blas) and ITK4.8. Since the parameter to be estimated are three (Fp, ADCslow and ADCfast) the computation requires more than three DWI images but a good result is obtained with 6 or more DWI images. The features concerned 5 main areas: intensity analysis, gray level co-occurrence, gray level run length, wavelet transform and the shape features of the region of interest (ROI). It is implemented in C++ stl 11 and take advantages from the "insight segmentation and registration toolkit" (www.itk.org) and the OpenMP API for parallel programming (http://openmp.org/wp). The computation is optimized for a pc with a minimum RAM of 4GB; multi-core CPUs is not mandatory but is preferable. FE is compiled under Linux, Windows and Mac. FE performs the analysis on any region of interest inside the field of view of the MRI volume which can be tumors, lymph nodes, or healthy tissues. For each ROI and image, an array of features is computed and exported as plain text. In particular, for each pair of patient / ROI, a single CSV file for the 12 images is prepared. The CSV file has 12 rows, corresponding to the images where the analysis is performed, and 641 columns, corresponding to the 641

features. The integration with the Fraunhofer software is guaranteed through their JSON file output.

Phenotipization Tool (PT) is a tool to perform unsupervised clustering to reveal possible clusters of patients with similar radiomic expression patterns. The main clusters of patients will be related with clinical outcome (survival) and clinical characteristics, looking for association with tumor stage, histology etc. To build a prognostic radiomic signature, the analysis will be divided in training and validation phases. To remove redundancy within the radiomic information, the best-performing radiomic features will be selected and combined into a multivariate Cox proportional hazards regression model for prediction of survival. The performance of the radiomic signature will be tested in the validation data sets. It is implemented in MATLAB and will be delivered as an executable file for Linux, Windows and Mac.

CT radiomics feature extractor tool is used to extract a set of imaging features from CT scans. The software expects a CT scan and a segmentation of the structure of interest, for which the radiomic features shall be calculated. The feature extractor tool uses DICOM for the CT image data and DICOM-RT for the segmentations, as input data. The output consists of a CSV file per tumor, with the extractor radiomic features.

3.7 Statistical models technology

The Statistical models (Model library and Model synthesis tool) are essentially an input-output function with an R backend where are develop the statistical functions. The R-backend will be capable of reading the data from a tab delimited text file. The input data (clinical, population, radiomic and genomic data) may either be available as joint vector/matrix file or in separate files. Each row in a matrix will be considered as a patient. The R-function will pre-process the data and compute the statistics. Output will be a tab delimited text file.

After the retrospective study, the models will be used as services that will access to the data through the APIs that AII are developing. Then, once they process the data, the outcomes will be send through web services. These outputs will be used in different ways: these will be directly represented in the CDSS when clinician ask for prediction, and also the outcomes will be stored in the Big Data Infrastructure to be available for a prediction analysis requested for a set of patients (these outcomes will be accessed by the tools through the APIs).

In the following schemas there are presented the different statistical models developed and integrated in the BD2Decide system.

In schema Figure 19, the model library diagram is shown.

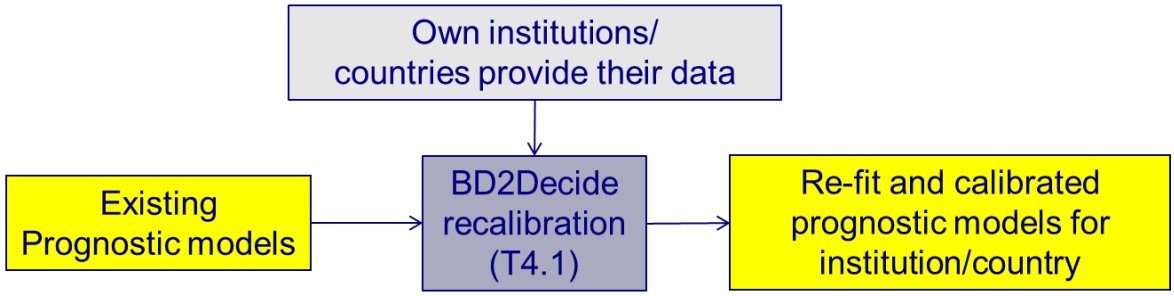


Figure 19 - Statistical models. Model library schema

Here in the schema Figure 21, the integration of the Model synthesis tool with the Model library is exposed.

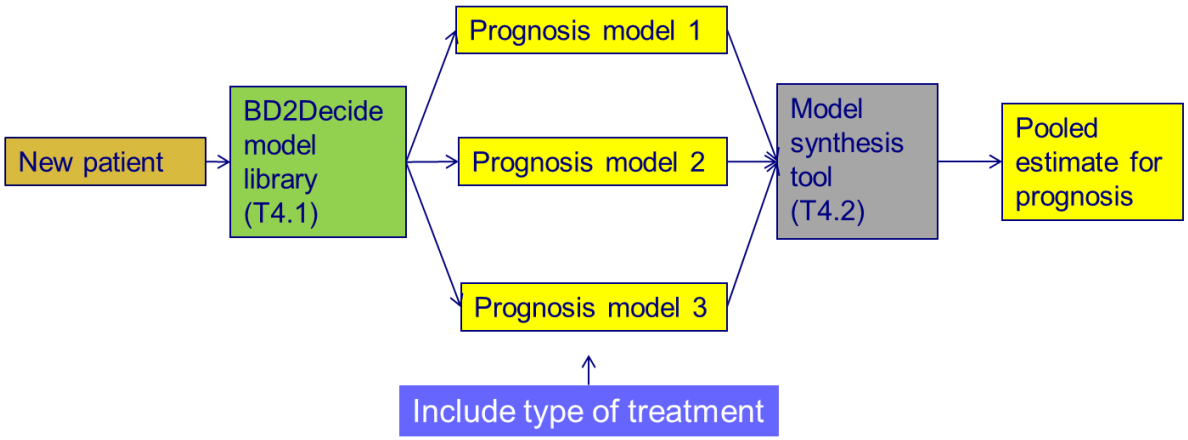


Figure 20 - Statistical models. Model synthesis tool schema

In scheme Figure 21, the Model updating diagram tool is presented.

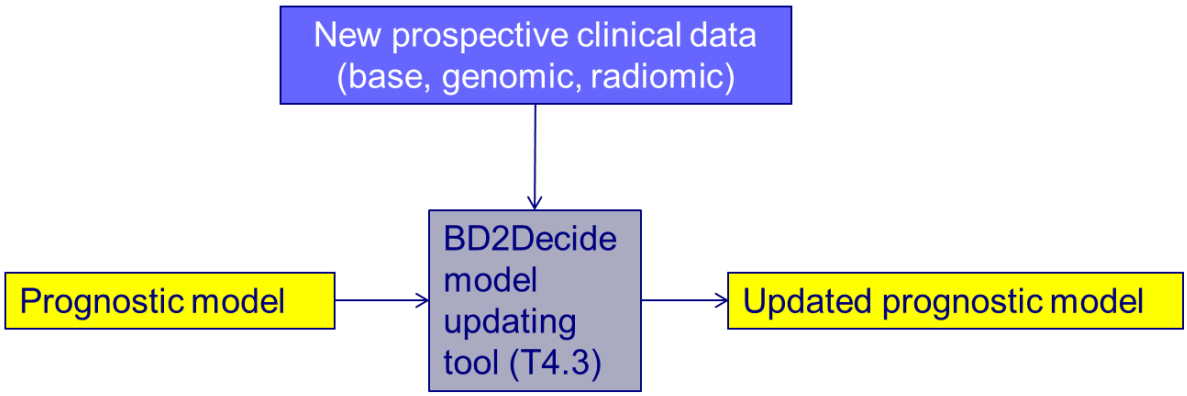


Figure 21 - Statistical models. Model updating tool schema

The last model developed (Cost-utility analysis tool) is the presented in the schema Figure 22.

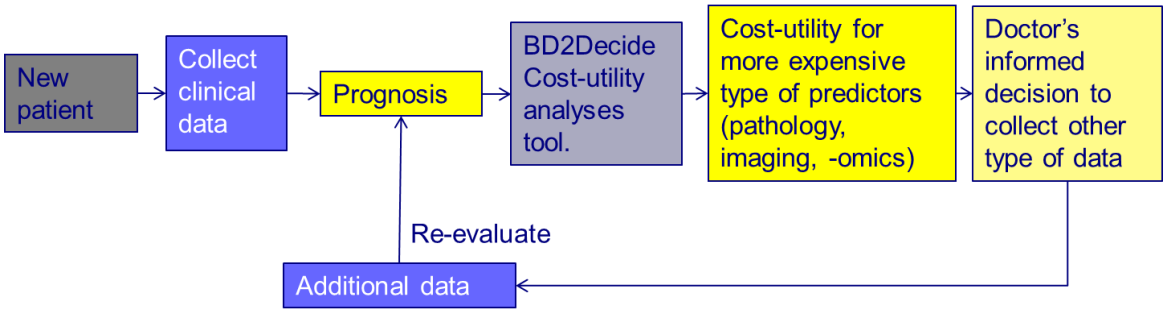


Figure 22 - Statistical models. Cost-utility analysis tool schema

4. UML DIAGRAMS

Unified Modelling Language (UML) is a standard visual modelling language which can be used for modelling processes and analyze and design software-based systems. In this section UML is used to support the definition of the BD2Decide components and to better understand the functionality, usage and interaction of each one. The way to represent these system models is through UML diagrams. There are some different kinds of diagrams, classified mainly between structure diagrams and behavior diagrams, although sometimes the last ones are divided also into interaction diagrams.

With the aim of clarifying the structure of the BD2Decide components, the UML diagram chosen in this document is the Class diagram. And to complete the definition of the tools also an Activity diagram of each application is represented. The last one is going to help to understand the behavior of each tool including the activities flows and dependencies between the different tasks.

4.1 Clinical DSS tool suite

This paragraph shows the class diagram and the activity diagram of the Clinical DSS tool suite.

4.1.1 Class diagram

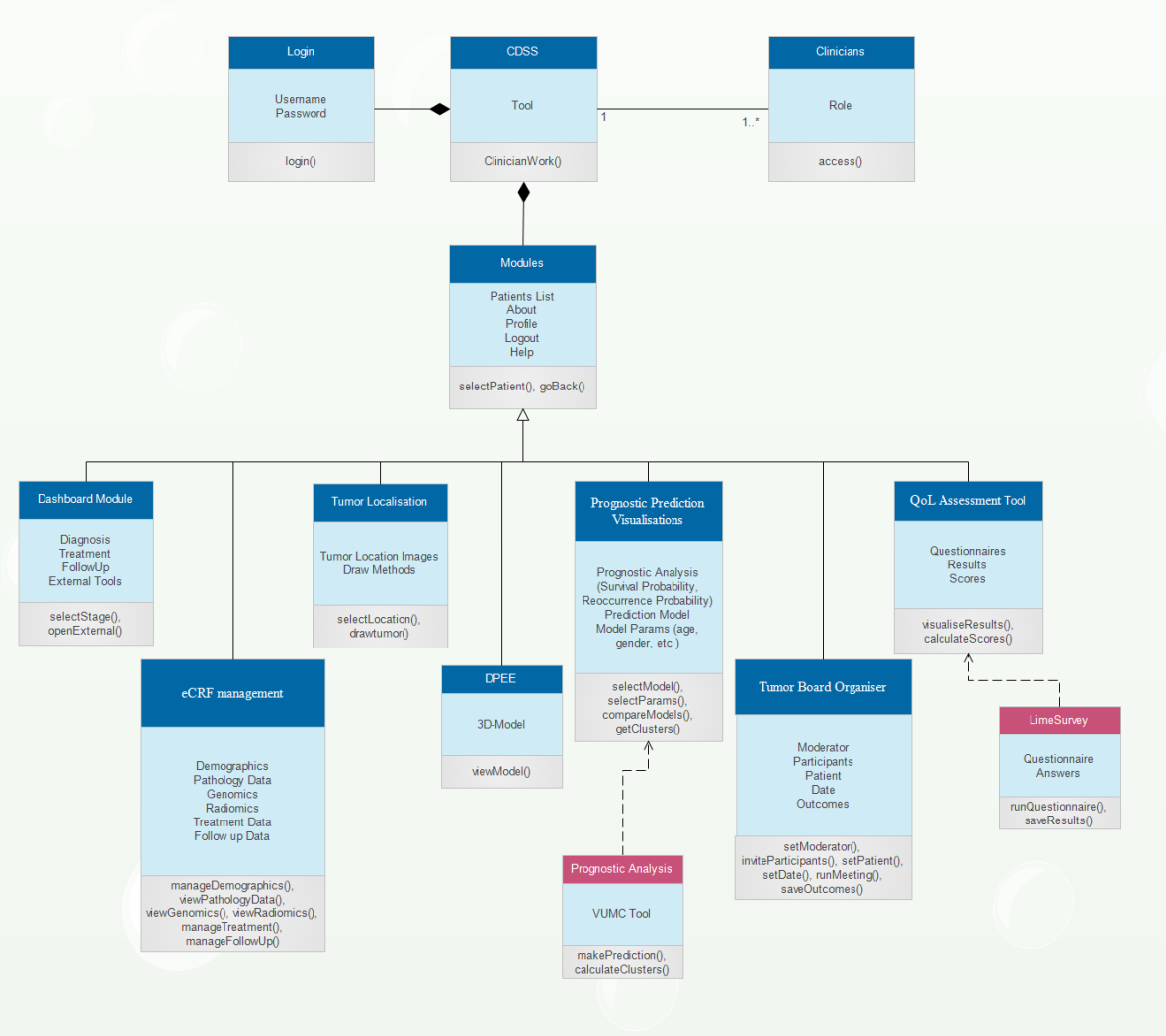


Figure 23 - CDSS class diagram

4.1.2 Activity diagram

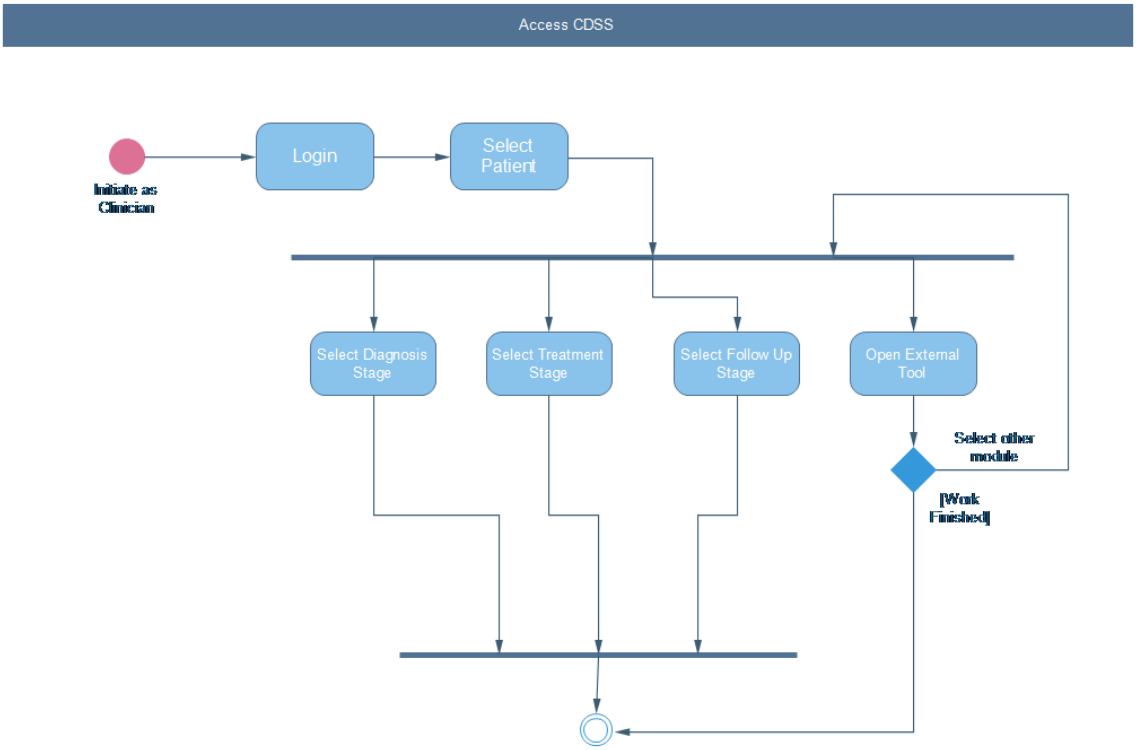


Figure 24 - CDSS activity diagram. Access

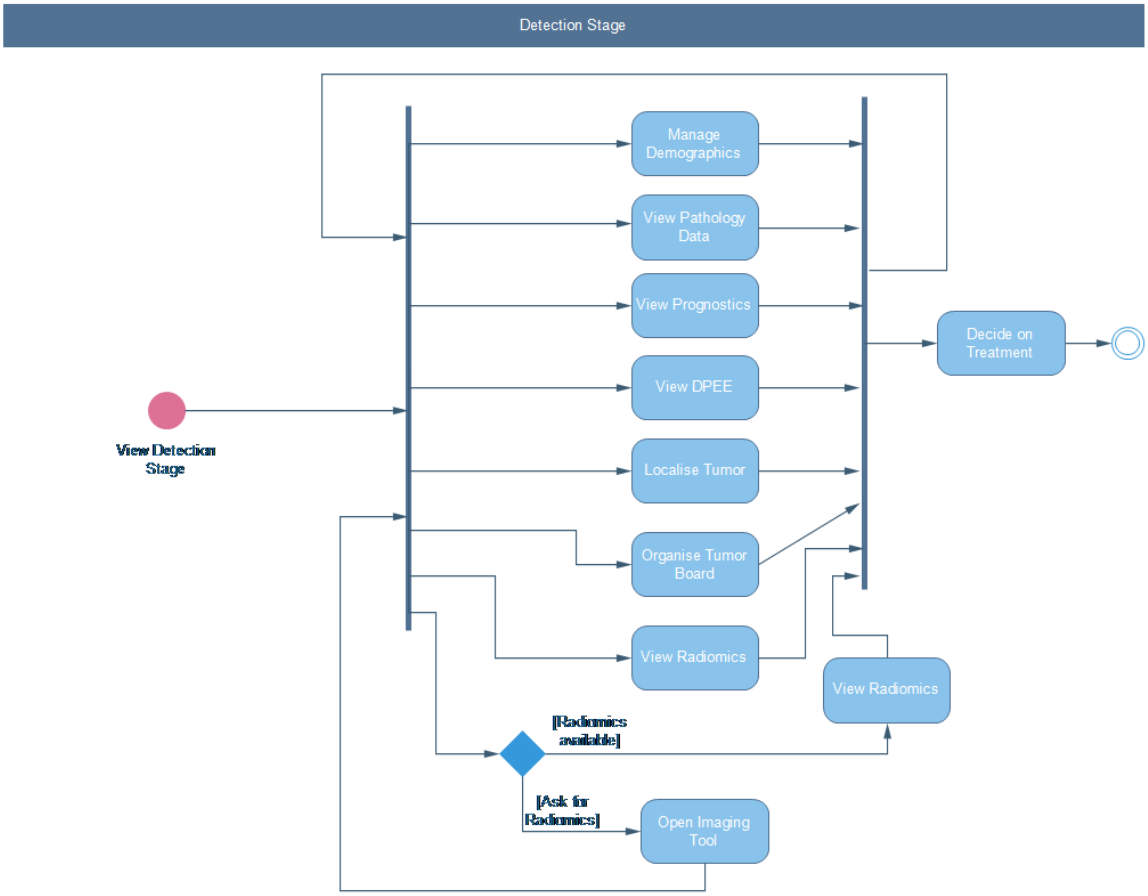


Figure 25 - CDSS activity diagram. Decision stage

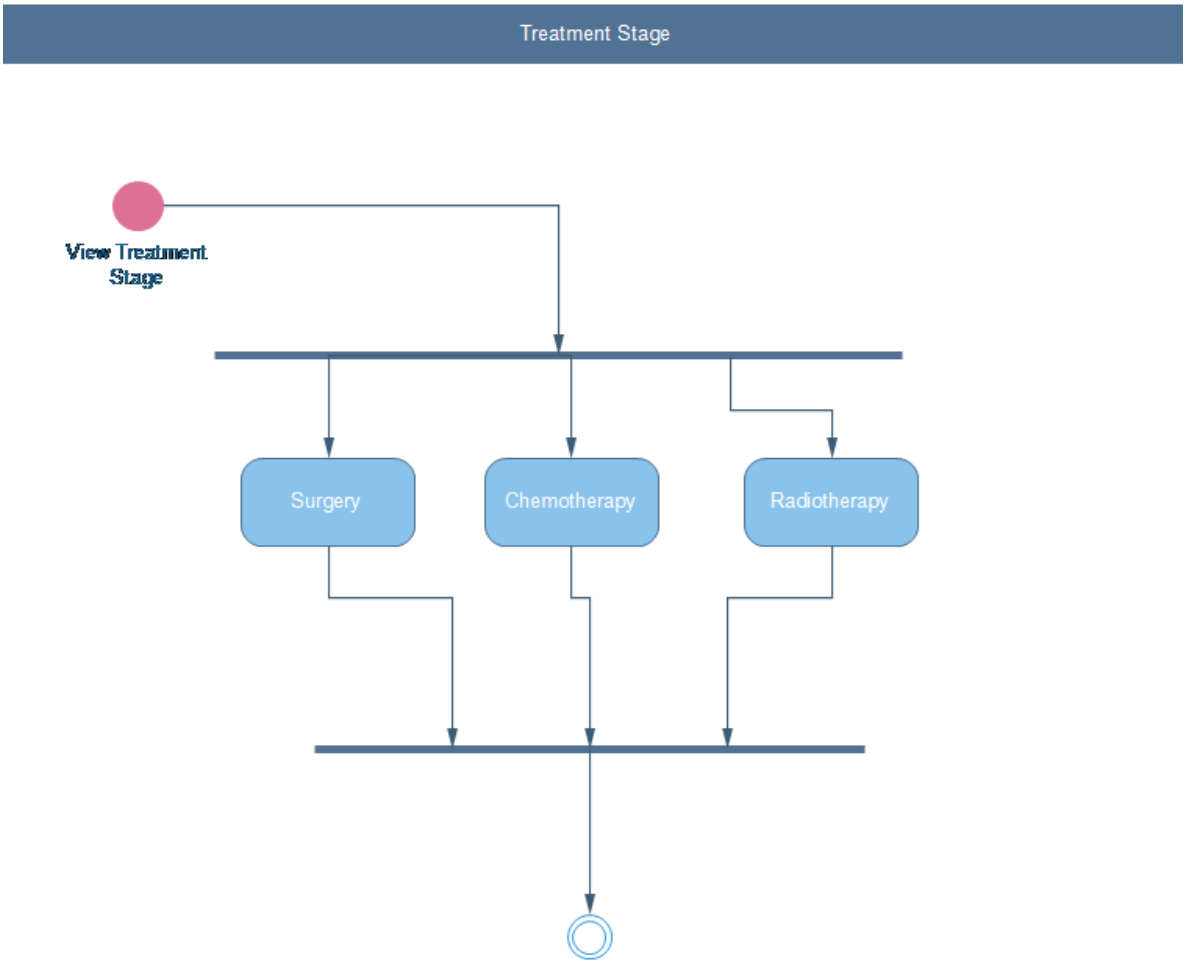


Figure 26 - CDSS activity diagram. Treatment stage

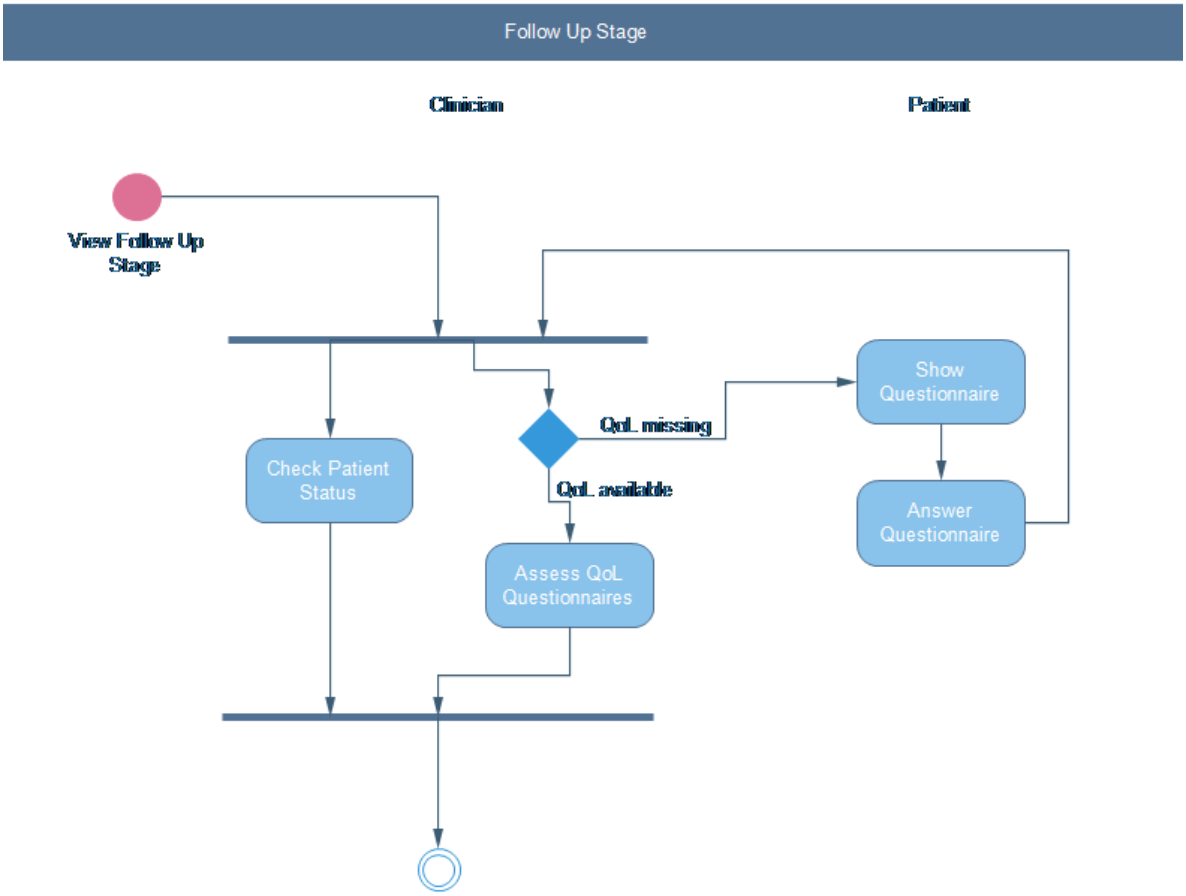


Figure 27 - CDSS activity diagram. Follow up stage

4.2 Visual Analytics Tool

Having introduced the types of diagrams used and the elements of each one, in this subsection the UML diagrams that models the researcher tool are presented.

Looking at the Figure 28, the class diagram shows all the modules involved in the researcher tool, and all other components related to that. Furthermore, all the relations between each class and the type of the relationship are represented: besides the composition links, the classes are differentiated by color: the light blue classes means to the user roles that will be available to use the VAT; the dark blue classes represent the modules contained in the tool (the functionalities available); the pink ones represent other BD2Decide components that are related in the researcher tool; and finally the orange class represents CDSS tool that are connected in some way with the VAT.

From Figure 29 to Figure 36, the activity diagrams are represented. In these, it is possible to see the interactions between the components of the Visual Analytics Tool, indicating the behavior that this tool carries out and the actions flow. Figure 29 shows the activity involved when a system administrator user access to the application, besides the Figure 30

shows the activity involved when the user role is a researcher itself. The rest of Visual Analytics Tool activity diagrams are the extension of the activities included in the Figure 30.

4.2.1 Class diagram

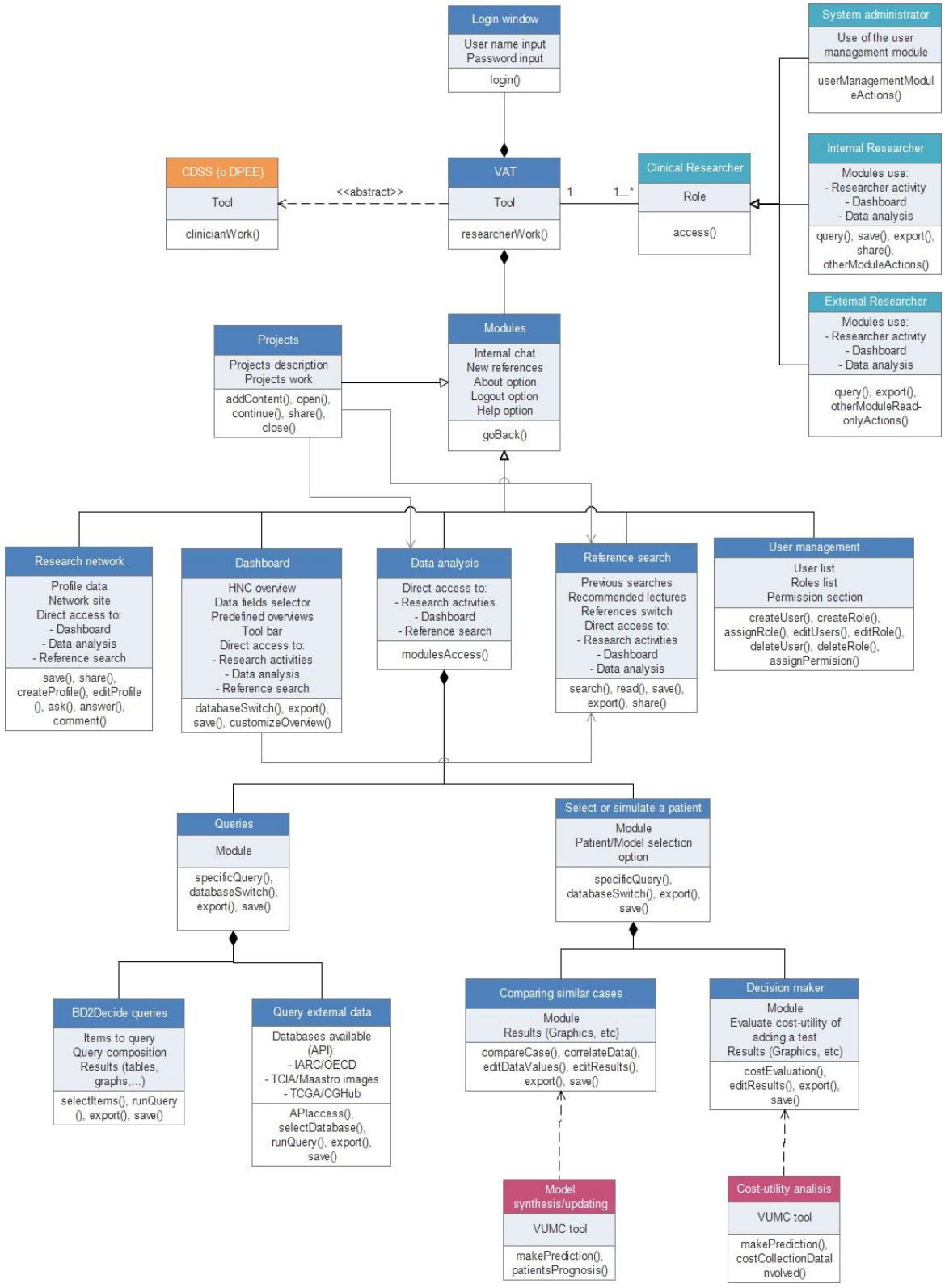


Figure 28 - Visual Analytics Tool class diagram

4.2.2 Activity diagram

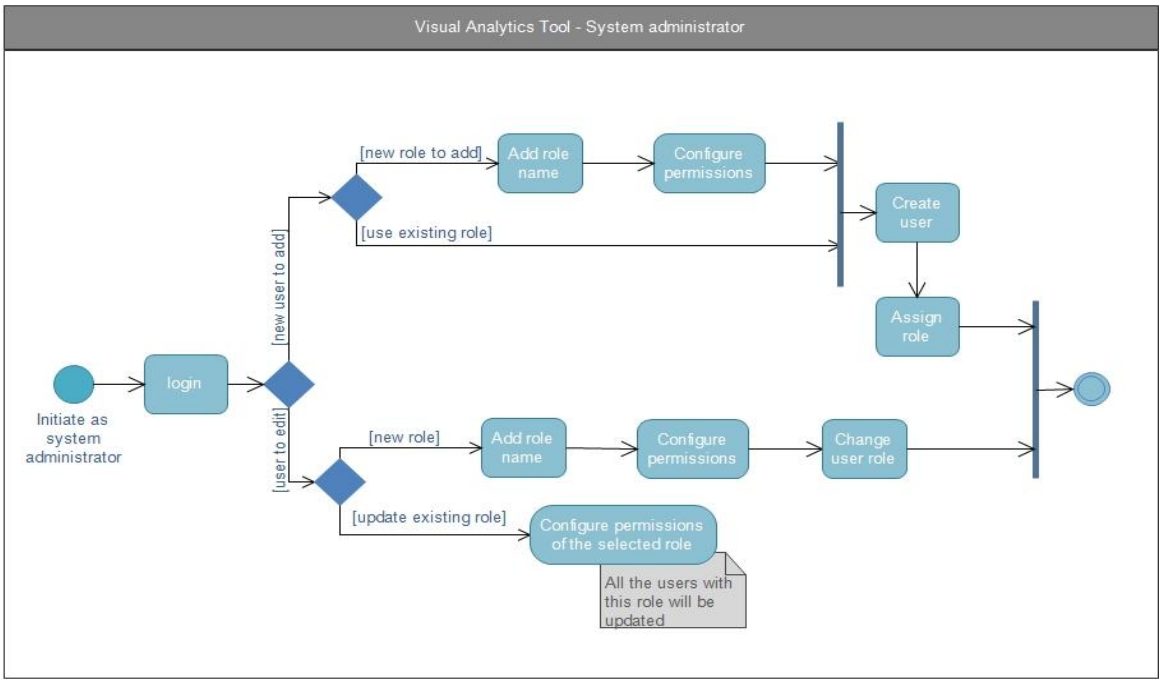


Figure 29 - VAT activity diagram. System administrator access

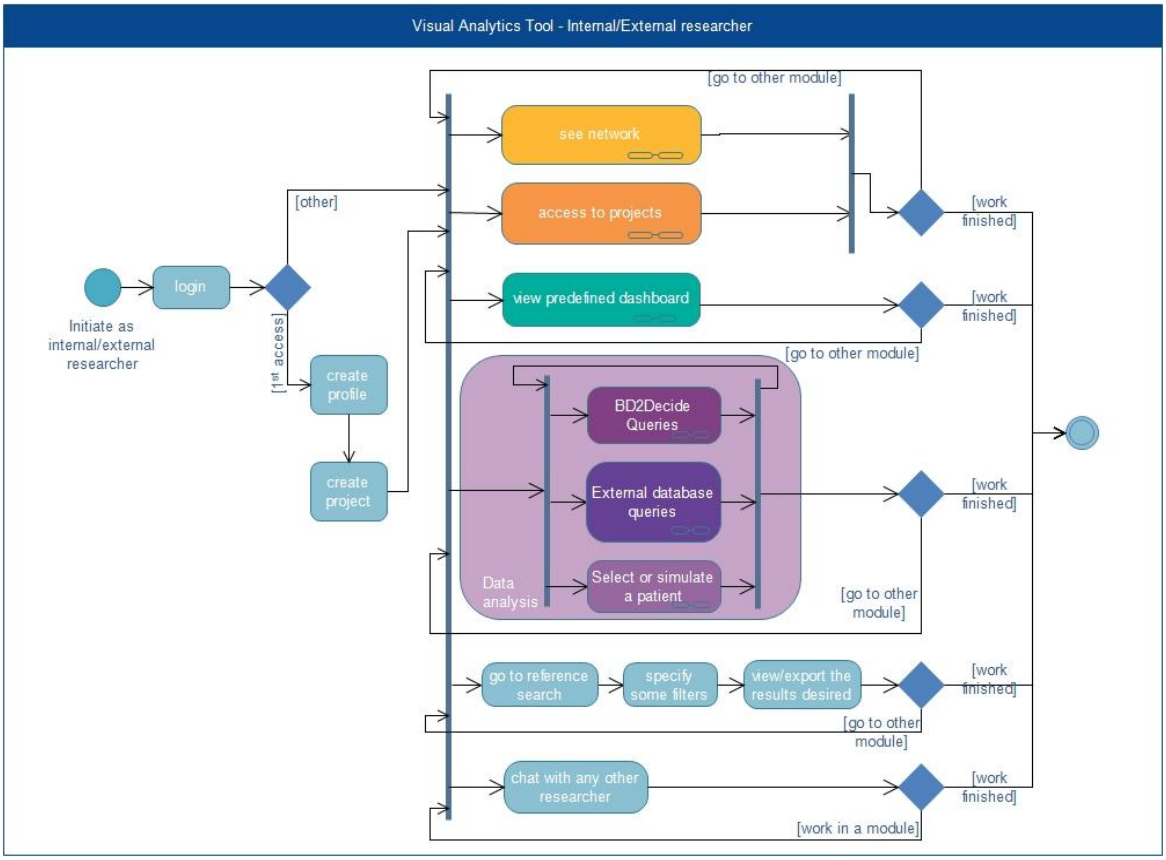


Figure 30 - VAT activity diagram. Internal/External researcher access

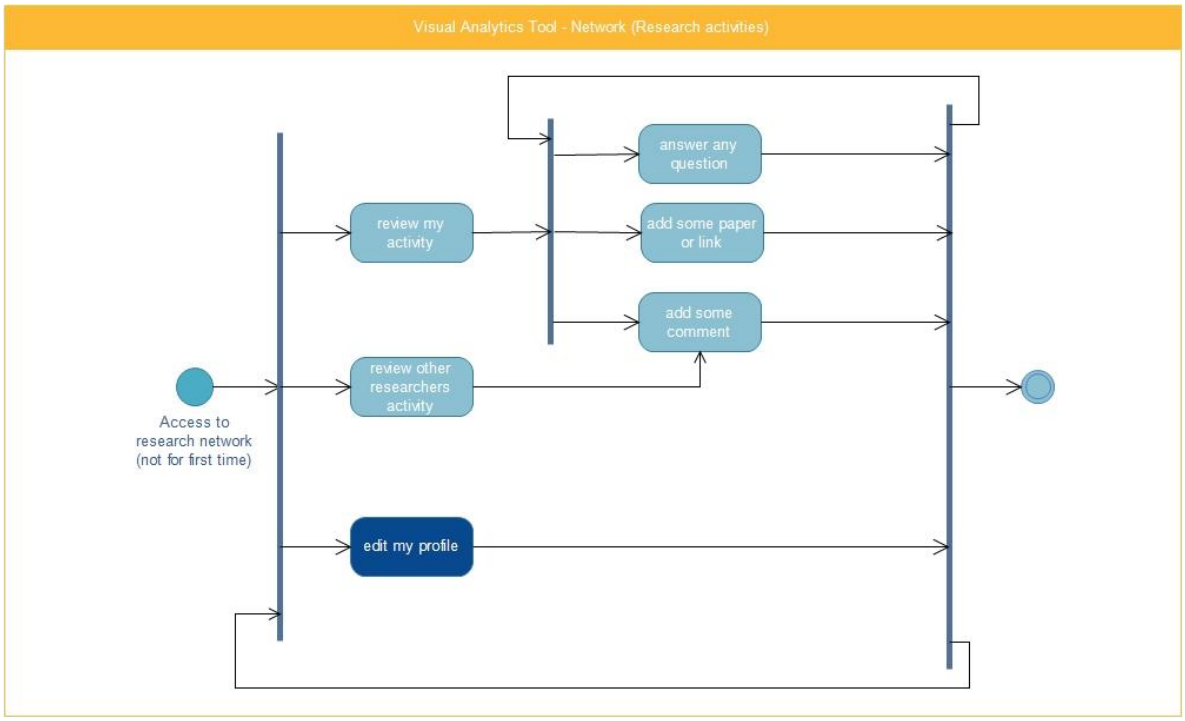


Figure 31 - VAT activity diagram. Researcher network

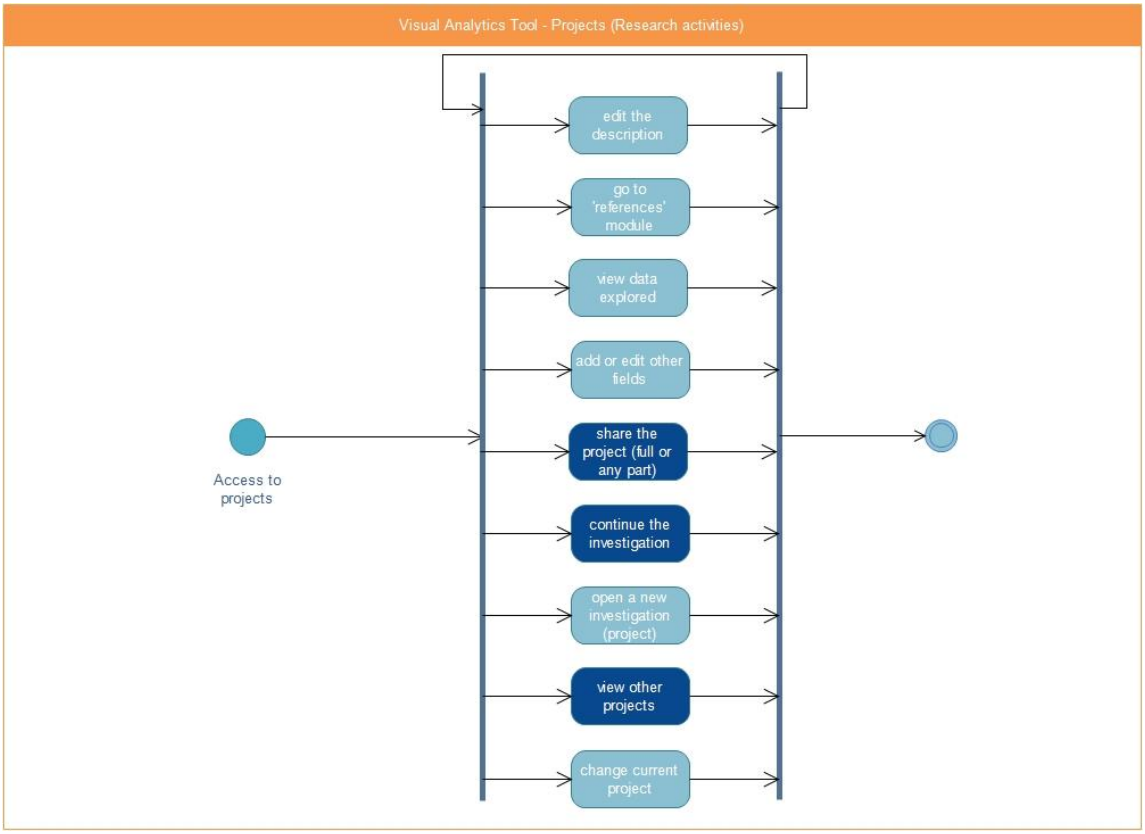


Figure 32 - VAT activity diagram. Researcher projects

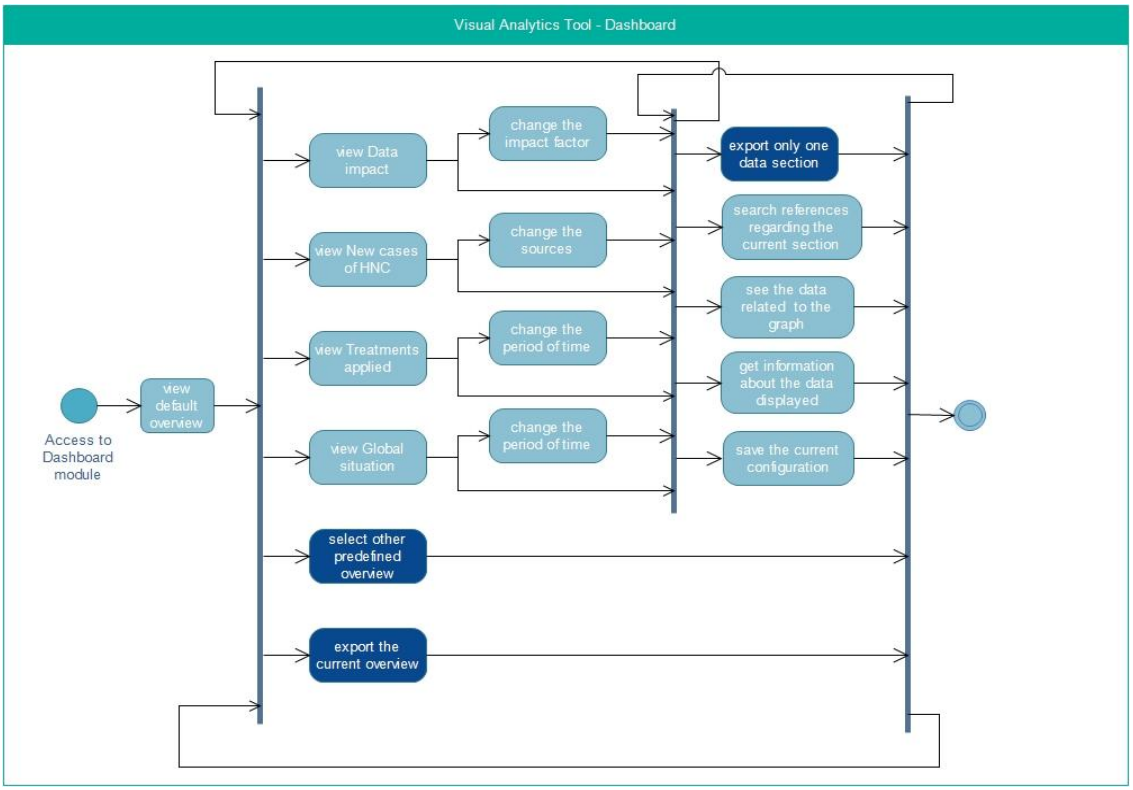


Figure 33 - VAT activity diagram. Dashboard

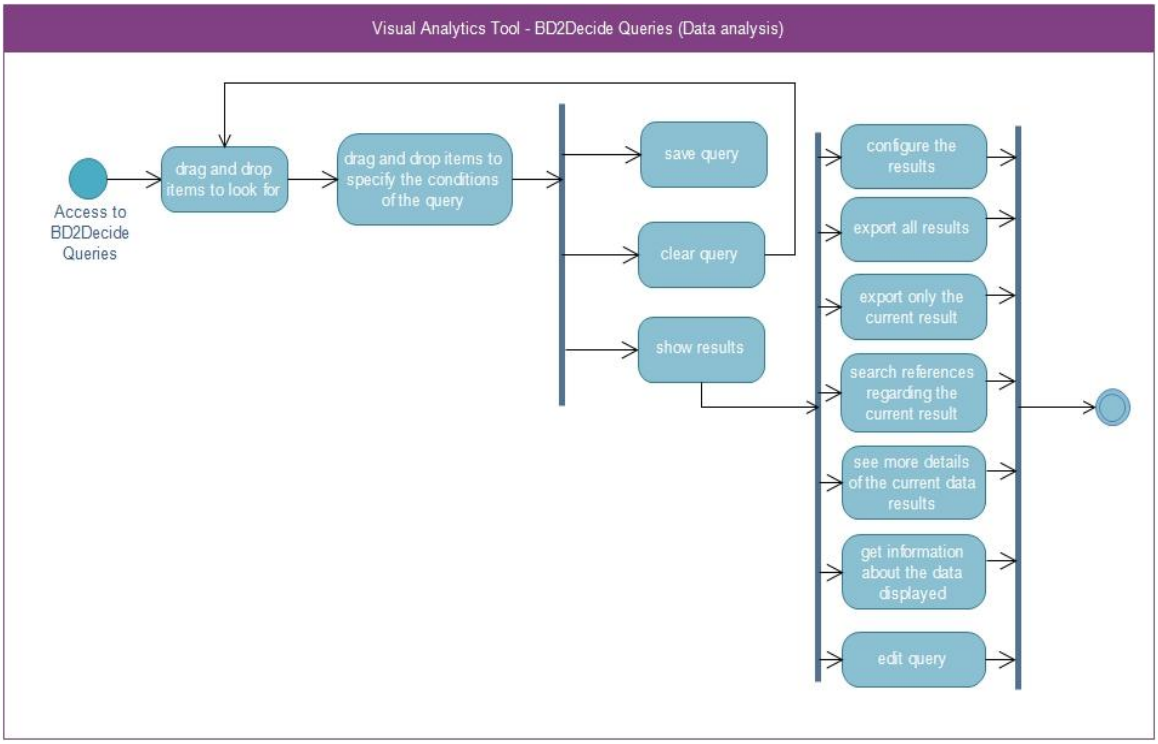


Figure 34 - VAT activity diagram. Query BD2Decide data

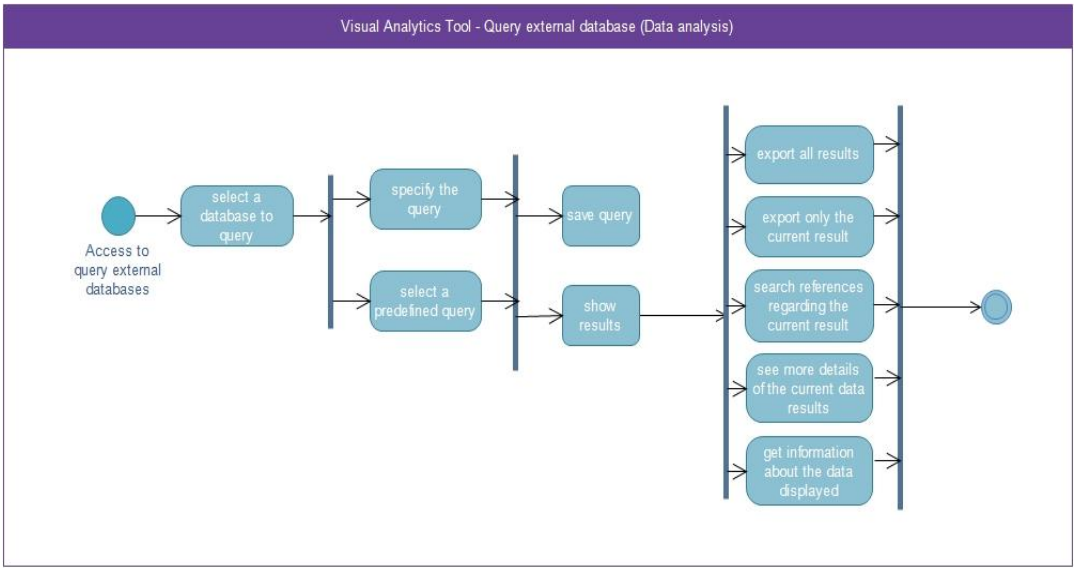


Figure 35 - VAT activity diagram. Query external data sources

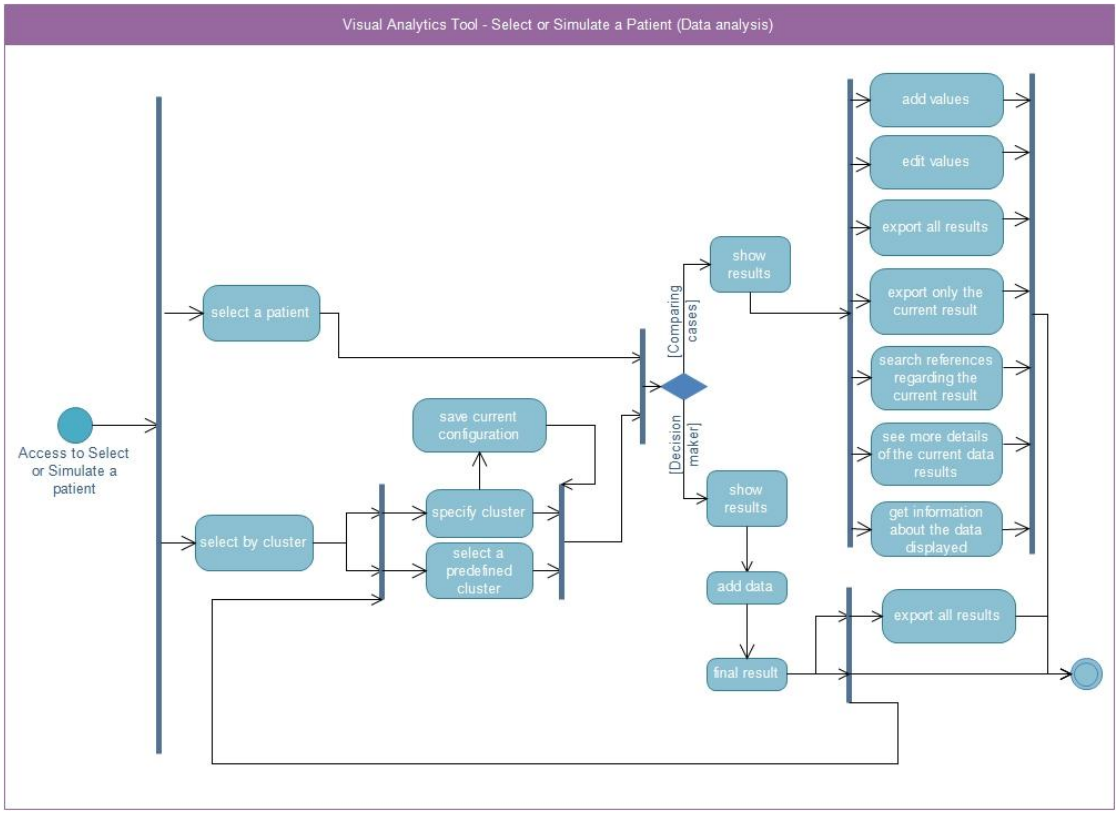


Figure 36 - VAT activity diagram. Select or simulate a patient (to Comparing similar cases or to go to Decision maker)

4.3 Interactive Patient's co-Decision Aid

In this section, the class and activity diagrams of the IPDA are presented.

4.3.1 Class diagram

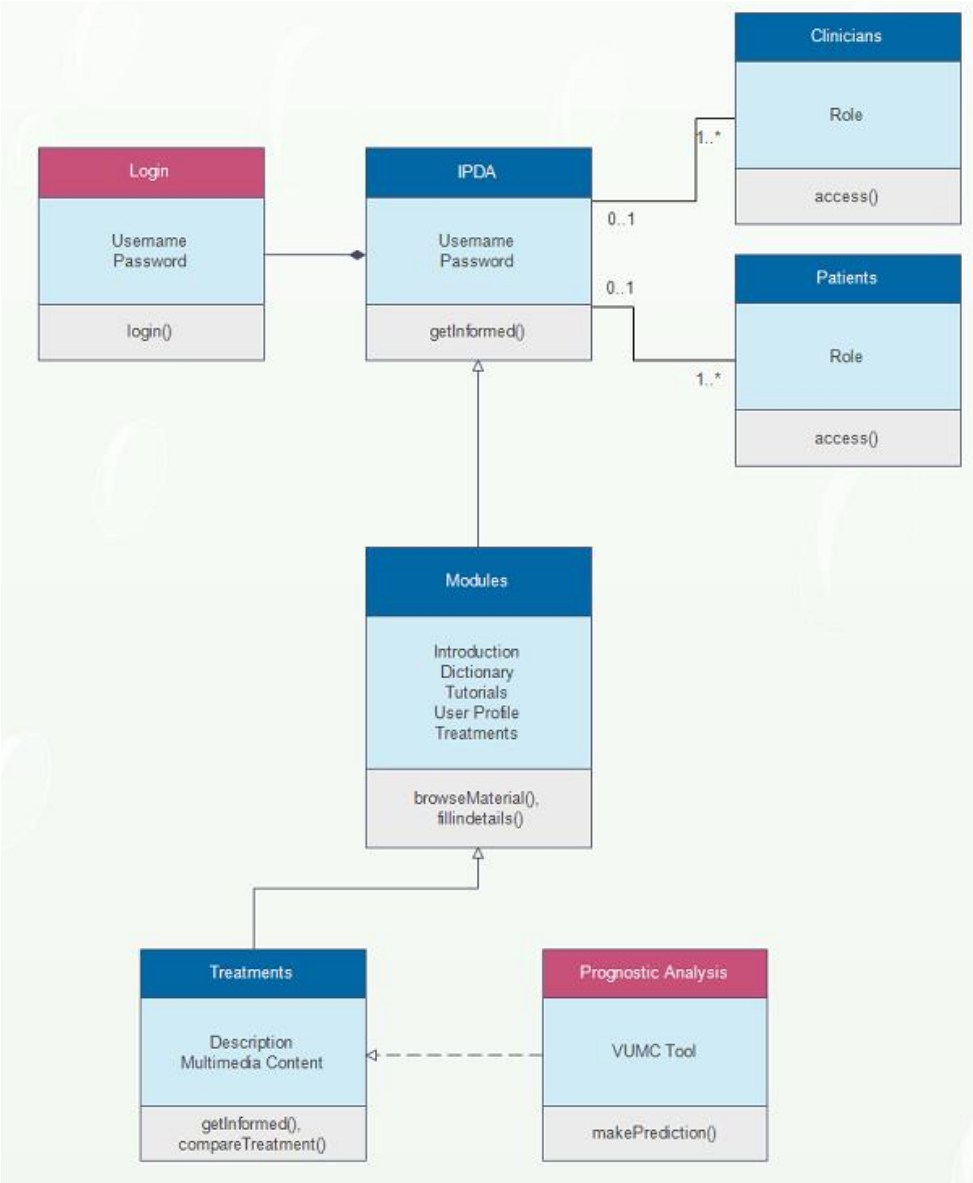


Figure 37 - IPDA class diagram

4.3.2 Activity diagram

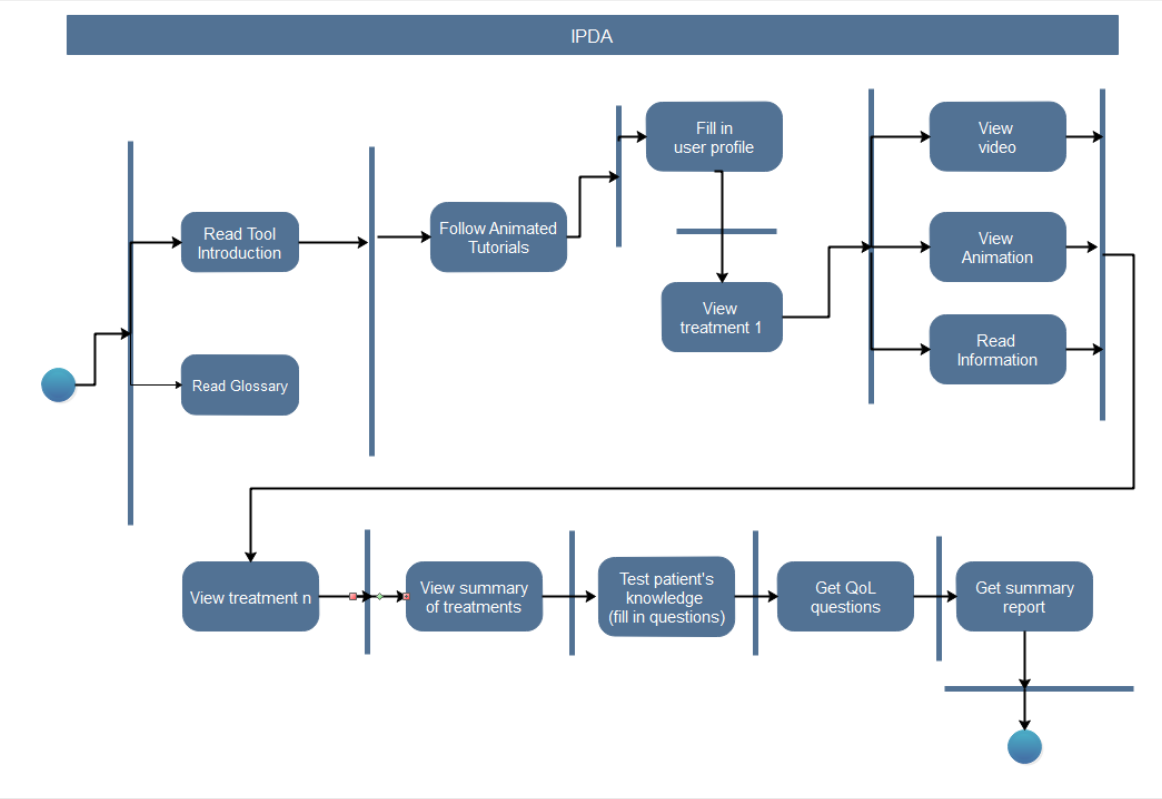


Figure 38 - IPDA activity diagram

4.4 Imaging models

In this chapter the UML diagrams related with the imaging treatment are presented. In the Figure 39, the class diagram models the structure of the image treatment, including the segmented and feature extraction system, and the radiomic extraction software products. In the Figure 40 the class diagram of the radiomic feature extraction by POLIMI is detailed. Finally, in paragraph 4.4.2 the activity diagrams related with the imaging models are shown.

4.4.1 Class diagram

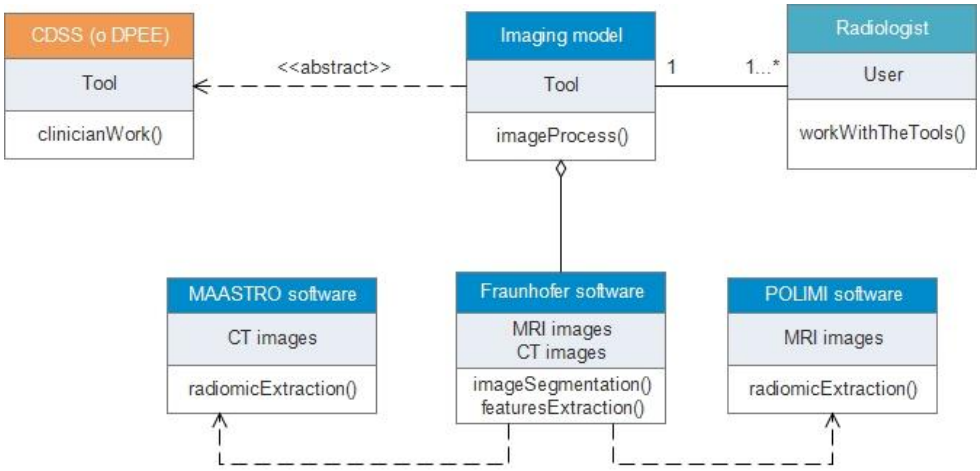


Figure 39 - Imaging model class diagram

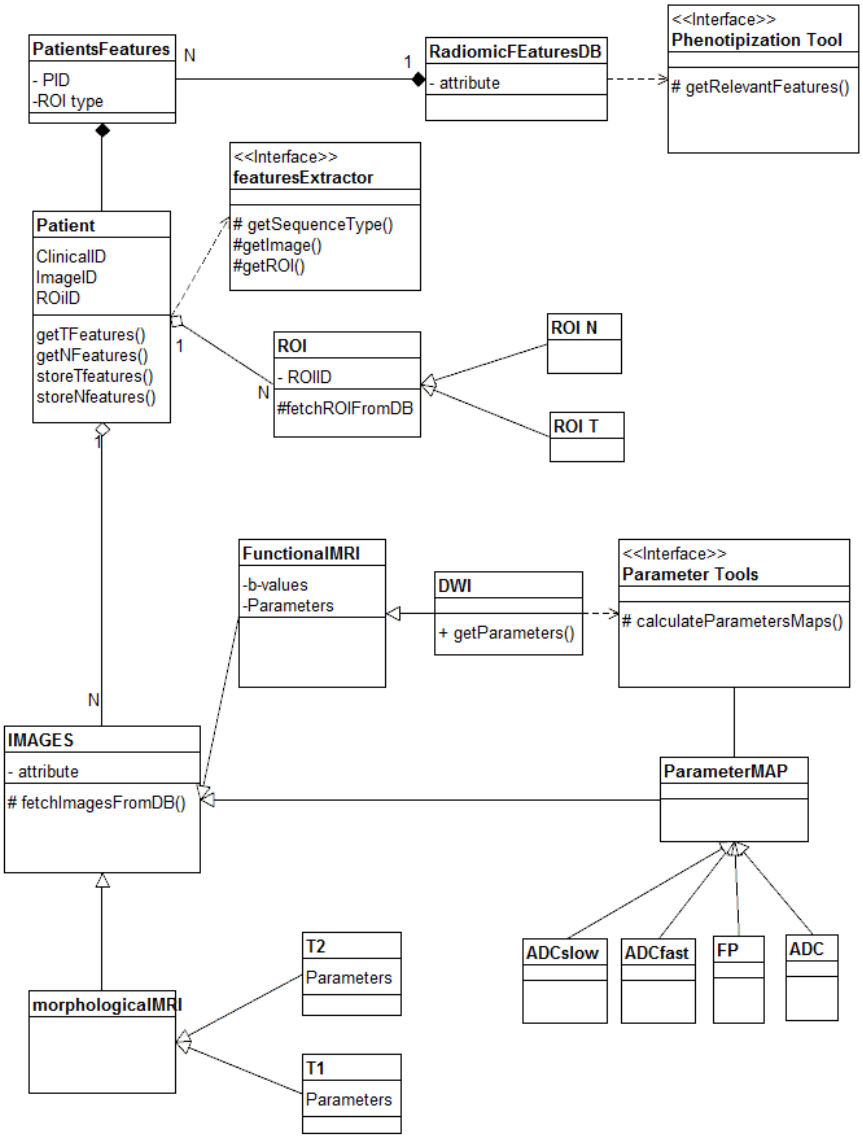


Figure 40 - POLIMI software class diagram

4.4.2 Activity diagram

Once the structure of the imaging model has been introduced, the following diagrams help to know the behavior of each activity involved.

In Figure 41, the activity related with the first phase of imaging processing, segmentation and feature extraction, is shown. After this flow, the processes represented in Figure 42 continues with the imaging process, extracting the radiomic features.

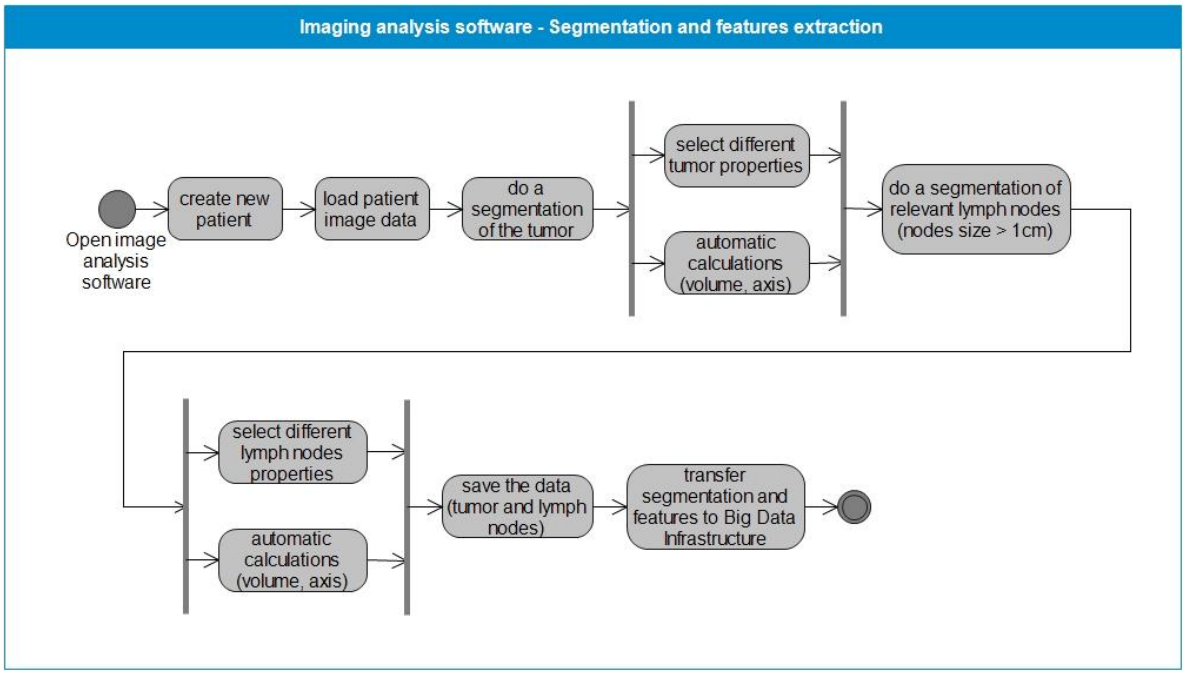


Figure 41 - Fraunhofer software activity diagram

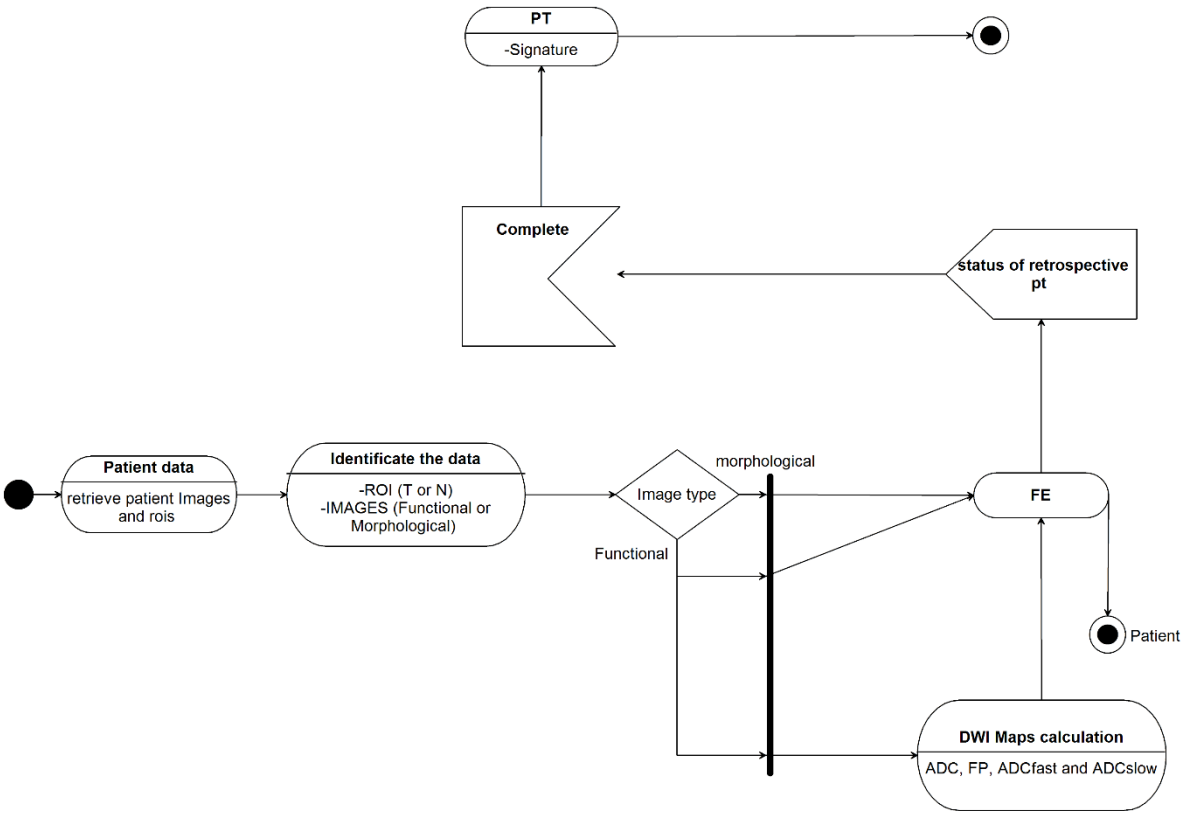


Figure 42 - POLIMI software activity diagram

4.5 Statistical models

In this paragraph the activity UML diagram of the 'Model library' and 'Model synthesis tool' is represented.

In Figure 43 all the task involved in the statistical processing are specified.

The user-specified parameters are: the desired level of confidence, the type of published model or model synthesis and the desired outcome (availability is model dependent), so each task ('Compute prediction and variances' and 'Compute confidence intervals') will perform a specific function depending on that. These functions details are not included in the diagram because they are mathematical functions.

4.5.1 Activity diagram

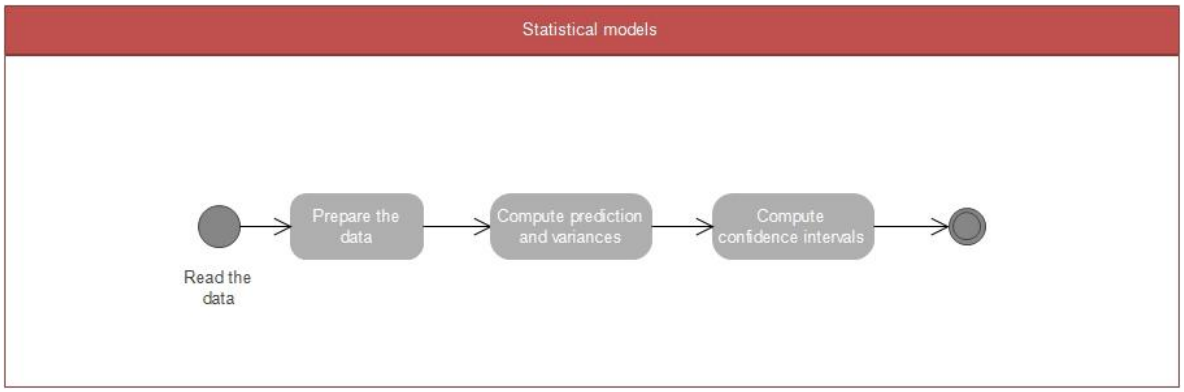


Figure 43 - Statistical models activity diagram

5. DATA FLOW DIAGRAMS

This section aims to clear up the usage of the data sources involved in the BD2Decide project. Below there is a list with all the data repositories taken into account within the project:

- IPRR. Integrated Patients' Records Repository is the repository for the patient data. All the eCRF that should be included in the BDI should be included in this repository, as listed next:
  - Demographic & Clinical data.
  - Risk factor.
  - Clinical T- and N-Characteristics.
  - Pathology data.
  - Chemotherapy.
  - Radiotherapy.
  - Surgery.
  - Tissue sample (excluding genomic data).
  - Follow-up (including 'Quality of life at Last Evaluation').
  - Toxicity.
  - QoL questionnaires (EORTC QLQ-H&N30, EORTC QLQ-H&N30, EuroQol EQ-5D) answers.
  - Segmented and radiomics features.
  - Genomic signatures.
- Prognostic models output.
- Cost reference. It contains cost information about treatments, medications and analysis or other test in order to estimate costs to include in the prediction. Cost-utility analysis tool needs to know this information.
- Population data (created by INT and ISS).
- External data sources (environmental, epidemiology, medication and lifestyle/behavior data; imaging data; and genomic data).
- Literature sources (Guidelines & References).
- Images
  - Original images [DICOM].
  - Segmented images (ROI) [nrrd].
  - Segmented features.
  - Radiomics data from CT and MRI.
- Genomics.
  - Raw data [BAM].
  - Genomic signatures.
- IdM. Identity Management is a database for handling identity management.

- Media. Multimedia database hosting the media items used in the co-decision aid tool.
- Digital Patient Models. Containing the data for the 3D digital avatar models (3D visualizer).

In the following subsections the interaction with each data source is specified within each tool. However not all data types are used in each BD2Decide component, so in each section the proper data sources are taken into account.

5.1 Clinical DSS tool suite

The following image (Figure 44) shows the CDSS data flow diagram.

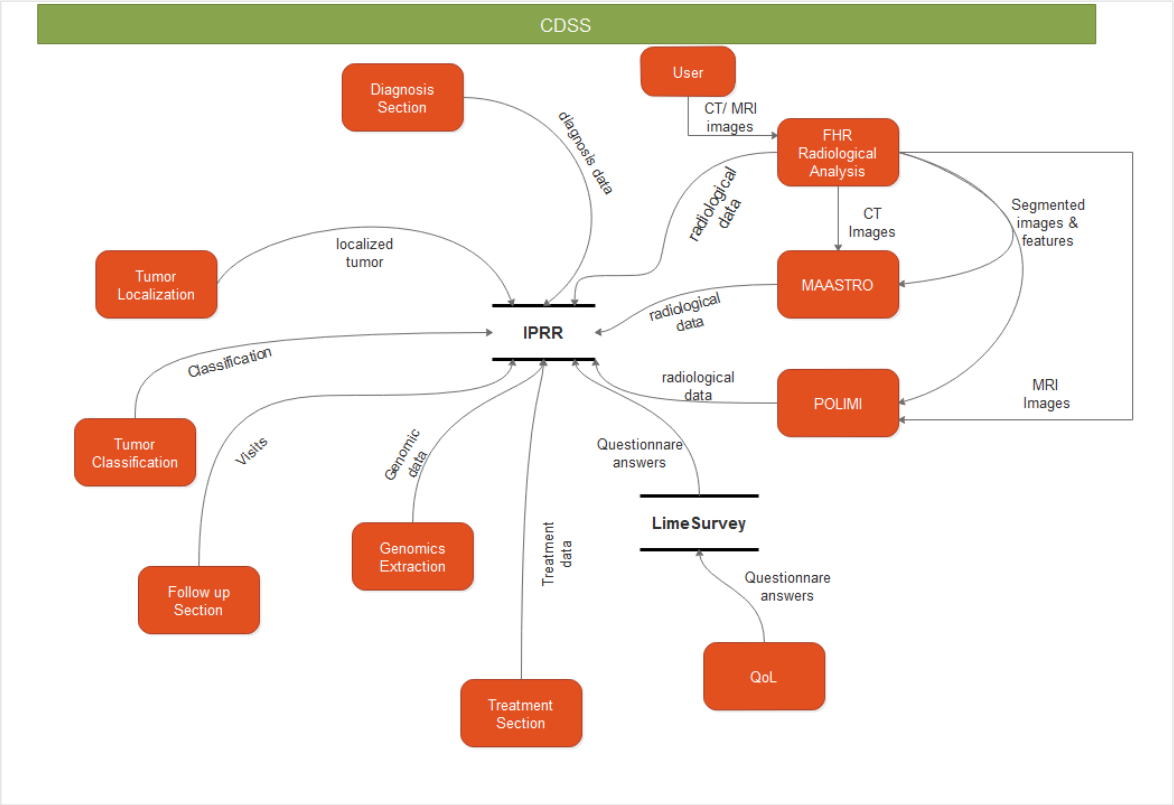


Figure 44 - CDSS data flow diagram

5.2 Visual Analytics Tool

In the Figure 45 the data interaction within the researcher tool is shown. Each element represented a specific meaning:

- The rhomboid represents modules contents into the visual analytics tool (in green) or other BD2Decide components (in pink tones).
- The rhombus represents decision actions.
- The rectangle shows the different data sources. The color of the data sources indicates the type of data: orange means external data sources, purple means center local storage data, blue is the patient data included in the BD2Decide storage environment, pink are the input/output of the statistical models and BD analytics, brown is the identity management database, and dark green means the researcher tool database itself.
- The arrows reflect the interaction between the data and the other elements included in the visual analytics tool. The more thickness characterize the direct link between data and the parts of the system and the thin ones symbolize only the path between elements to identify the relations. Direct link means that the data is used in the corresponding module to get the proper outcomes and reach the goals of the functionalities.

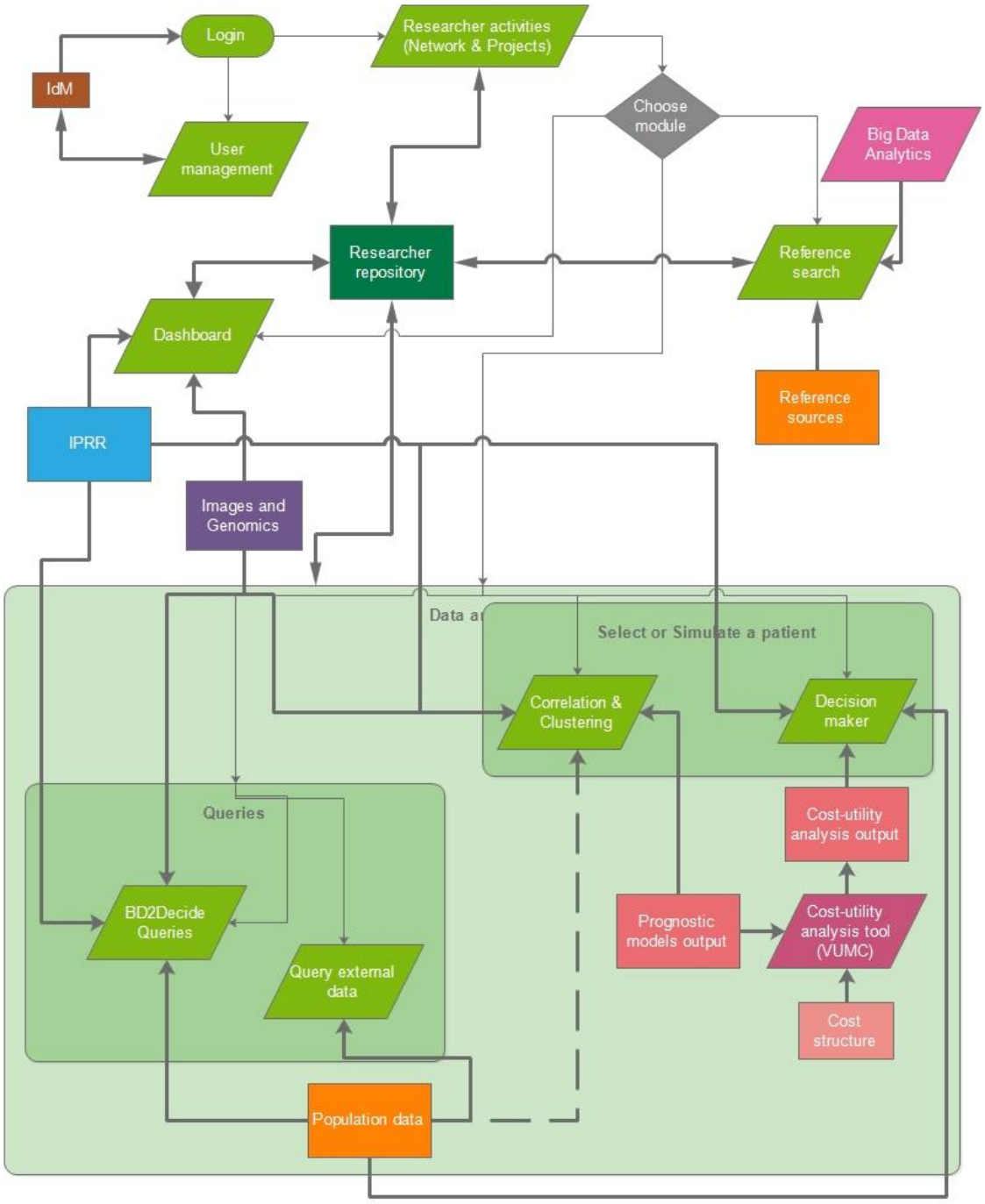


Figure 45 - Visual analytics tool data flow diagram

5.3 Interactive Patient's co-Decision Aid

Figure 46 shows the data flow diagram of the IPDA. The patient provides his/her profile including, name, hospital and stage of cancer (T3 or T4). Taking this input into account the IPDA presents only the information that is relevant for the profile.

In the IPDA the patient gets information about the treatment options (multimedia information and text). The patient can answer a knowledge test in the IPDA and chose his/her preferences regarding quality of life and treatment experience.

As output the IPDA provides a report which include the knowledge test results, preferences regarding QoL and treatment experience, as well as additional questions the patient may want to ask to the doctor.

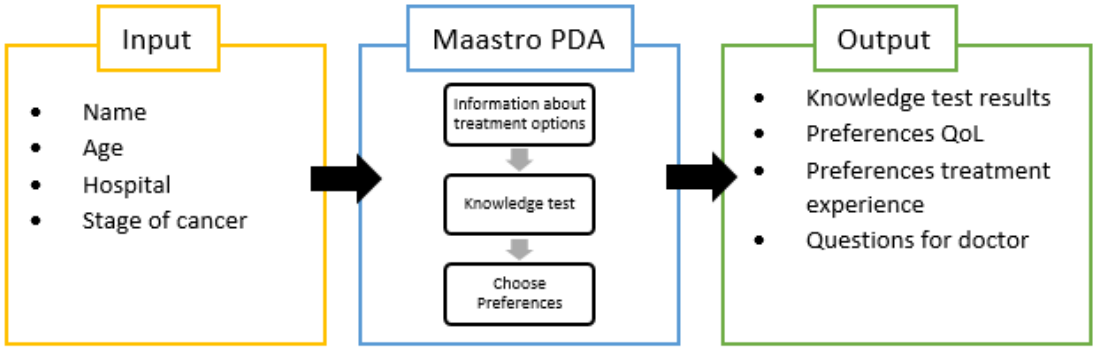


Figure 46 - IPDA Inputs and Outputs

The workflow of the IPDA is shown in Figure 47.

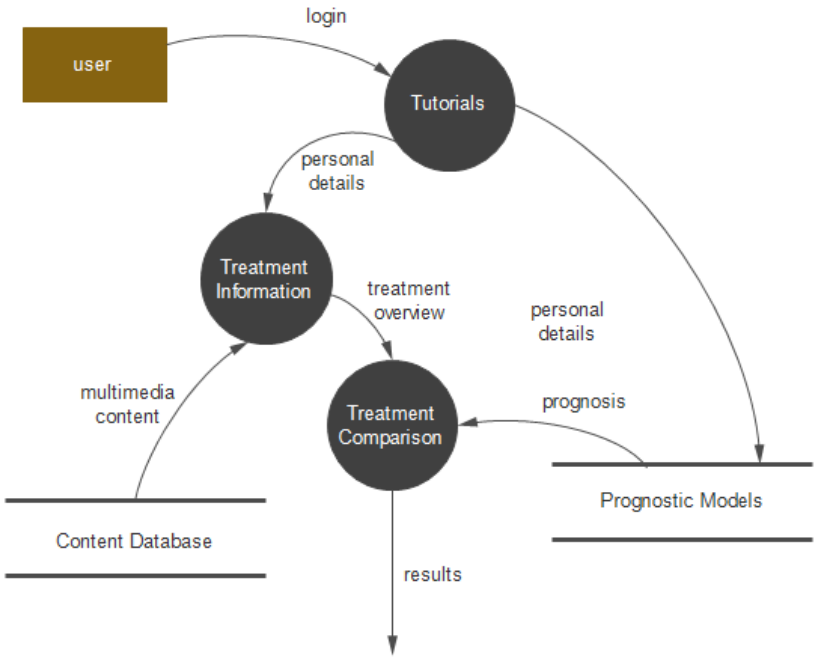


Figure 47 - IPDA data flow diagram

5.4 Imaging models

This section will focus on the data flow between the different image processing tools. The flow chart (Figure 48) is divided into 2 different locations. The hospital locations of Düsseldorf, Amsterdam, Parma and Milan are grouped together, because the workflow will be identical in these locations. In MAASTRO the workflow will be slightly different, therefore they have a separated location within the chart. The workflow in former hospitals is as follows. First the Fraunhofer image analysis tool is used. It retrieves the raw image data from a local NAS. The images are processed and the segmentations are stored on the local NAS. Additionally, the segmentations and extracted features are transferred to the BDI. Second the POLIMI image extractor is run. It retrieves the raw images from the NAS as well, as well as the segmented images, generated by the Fraunhofer image analysis tool. The POLIMI radiomics feature extractor, calculates the features and once done, sends them to the BDI as well. This is workflow for the 4 above mentioned hospitals.

At the clinical location in MAASTRO the above described steps are run as well and in addition the MAASTRO radiomics feature extractor on the raw image data of all hospitals. To accomplish this, the raw images and segmentations are loaded from the BDI. Then the radiomics features are extracted and finally send back to the BDI.

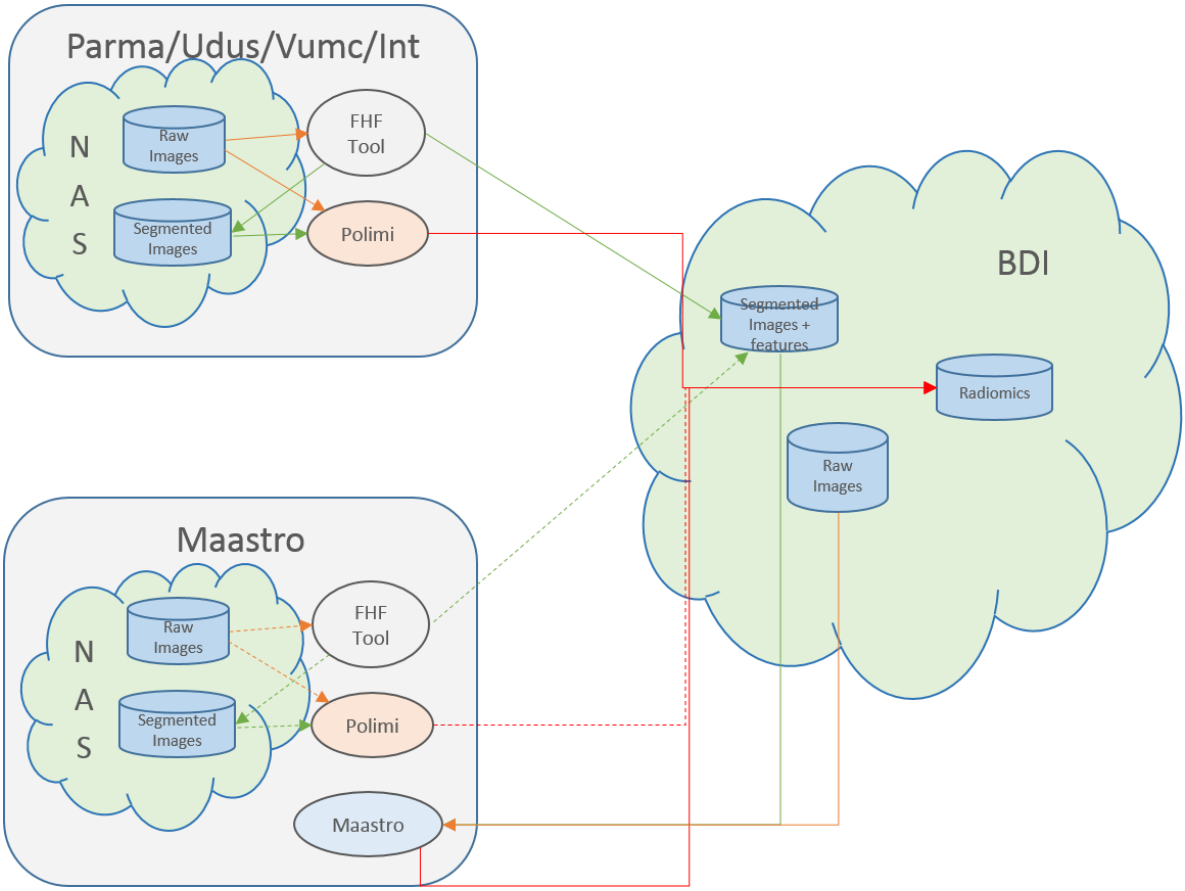


Figure 48 - Imaging data flow

FE takes as input morphological images, functional images and parameter maps, computed from the functional images as well as the ROIs. FE gives as output patient's features computed on all images and stores them in database (disk). When all patients' features are computed, PT takes them as input along with clinical outcome and gives as output a subset of features (signature) that are significantly correlated to the clinical outcome. See Figure 49.

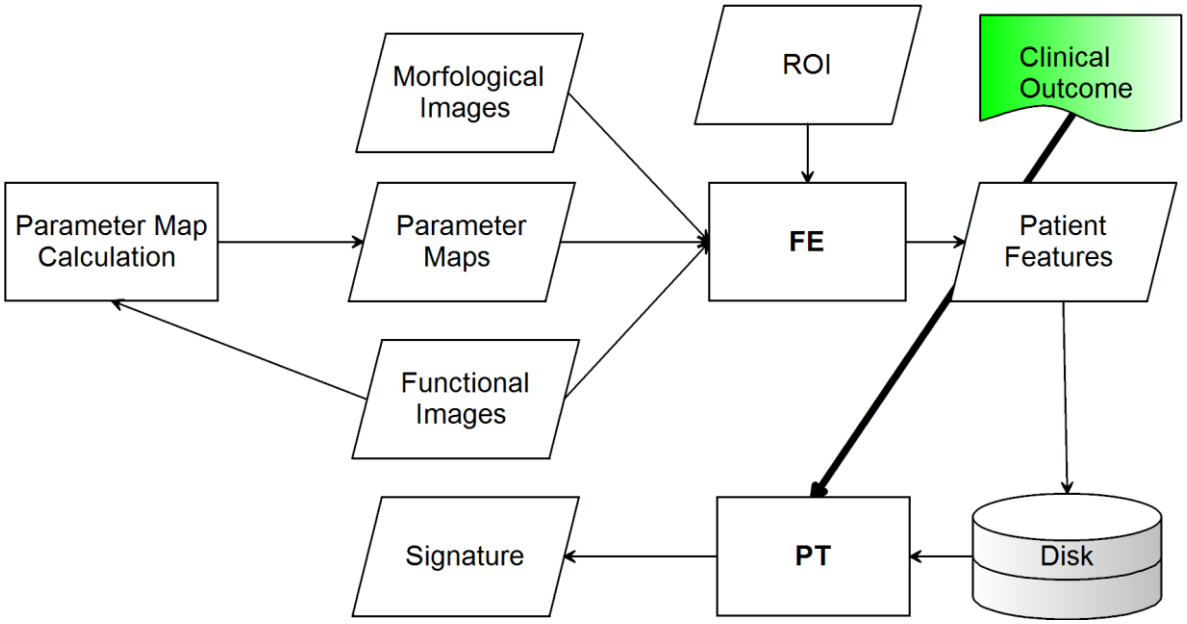


Figure 49 - MRI radiomic feature extraction (FE and PT process) data flow

The radiomics feature extractor uses CT image data and segmentations of the tumor as input (Figure 50). The tool calculates the radiomic features of the tumor. When the data of all patients is processed, a radiomic signature of the tumor can be learned. The output of the tool, are separate feature files per patient/tumor. And a radiomic signature of the tumor, which can be used for future patients.

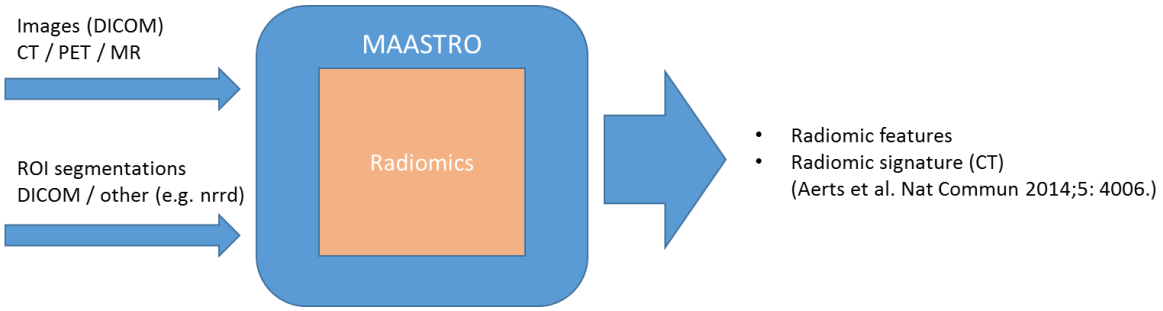


Figure 50 - Oncoradiomics data flow

5.5 Statistical models

In the Figure 51, the data flow of the statistical model is presented. The model predicts the outcome and confidence bounds for one new data point or a sample of *n* new data points from *p* predictors (*Output*). The predictors are taken from the clinical, genomic, radiomic and population data (*Input*) as mentioned in previous sections. The user may specify which type of model is to be used (defaults will be defined). Furthermore the level of confidence and the desired outcome (e.g. survival probability or hazard) can be specified. After that,

the results will be stored in the 'Statistical models output' database within the BD2Decide system and represented in the corresponding tool (VAT or CDSS).

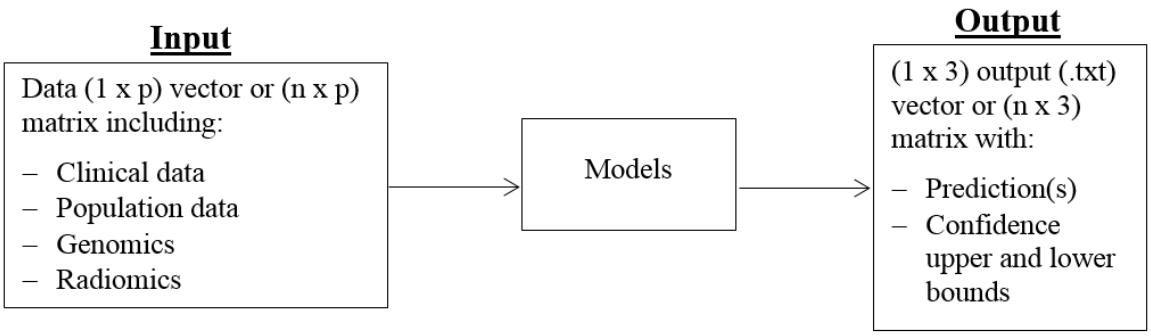


Figure 51 - Statistical models data flow diagram

6. CONCLUSIONS

This document, together with the deliverables D1.2, D4.1, D5.2, D6.1 and D7.2, has allowed the consortium to reach Project Milestone 2, *Technical and functional framework implemented:* all the key BD2Decide elements have been defined and described, including the BD2Decide components' interaction. However, as the definition of the core components of the DSS will take place during year 2 (prognostic models, feature extraction, big data analytics, knowledge management systems and the ontology) the definition of the final architecture will be finalized, accordingly.

The development of the design follows an iterative cycle: further design, deployment and validation phases are needed. These phases will be evaluated under the framework of Task 8.3: *Overall technical validation*. In particular, the following aspects will be validated:

1. Adherence to users' needs.
2. Level of integration of all modules.
3. Standardization of components and interoperability.
4. Data protection, security and maintenance.
5. Response times and accessibility.
6. Reliability.

Such aspects were taken into account also during the definition of the system architecture.

7. APPENDIX

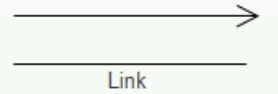
A brief description of the diagrams included in this section are explained below in order to understand the next sections.

Class diagram

This diagram shows the structure of the designed component as related classes and interfaces, with their features, constraints and relationships.

The elements of these diagrams are:

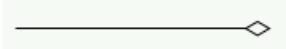
- Classes. A class is represented as a rectangle and it is divided into three areas:
  - Name of the class.
  - Attributes. These are the features of the corresponding class.
  - Actions. These are the operations that the proper class is able to do.
- Relation links, which represent the relation between the classes connected. These can be of different types among their meaning:
  - Association. These refer to static relations between the classes. The main symbols used to represent these links are:



    When a note appears in the association is to mark the number of instances of a class that can be linked to another.

  - Generalization. These indicate a directed relationship between a more general class (superclass) and a more specific classified (subclass). It is possible also to name it as Inheritance. Each instance of the specific classifier is also an indirect instance of the general classifier. This link symbol is:

  - Aggregation. This link represents association between a property and one or more composite objects which group together a set of instances. The classes related with this link are dependent, although the subclasses are part of the superclass. The symbol representing that relation is:



  - Composition. This relation is a ‘strong’ form of the aggregation. This relations create a link completely dependent between the classes related. The superclass cannot exist without the subclass. The symbol representing this relation is:

- Dependency. This is a direct relationship which is used to show that some UML elements or a set of elements requires, needs or depends on other model elements for specification or implementation. The symbol associated to this link is:

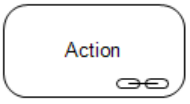
  Also it is possible find this symbol with the tag 'abstraction' which relates two elements representing the same concept but at different levels of abstraction.

Activity diagram

On the other hand, the activity diagram shows flow of control or object flow with emphasis on the sequence and conditions of the activity course.

The elements of these type of diagrams are:

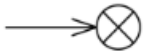
- Actions. These elements represent a single task within the activity. The set of actions sets up the activity that represents a behavior.
  Sometimes an action description needs another activity diagram, and it is represented including a symbol like crossed rings:

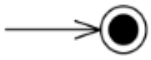
- Controls. These are activity nodes used to coordinate the flows between other nodes. Controls can be:
  - Initial node. It is a control node at which flow starts when the activity is invoked.

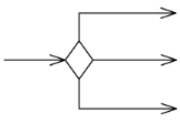
  - Flow final node. It is a control final node that terminates a flow. It destroys all tokens that arrive at it but has no effect on other flows in the activity.

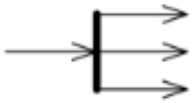
  - Activity final node. It is a control final node that stops all flows in an activity.

  - Decision node. It is a control node that accepts tokens on one or two incoming edges and selects one outgoing edge from one or more outgoing flows.

- Fork node. It is a control node that has one incoming edge and multiple outgoing edges and is used to split incoming flow into multiple concurrent flows. Fork nodes purpose is to support parallelism in activities.



- Join node. It is a control node that has multiple incoming edges and one outgoing edge and is used to synchronize incoming concurrent flows. Join nodes purpose, as fork node, is to support parallelism in activities.