



Grant Agreement no. 777167

BOUNCE

Predicting Effective Adaptation to Breast Cancer to Help Women to BOUNCE Back

Research and Innovation Action

SC1-PM-17-2017: *Personalised computer models and in-silico systems for well-being*

Deliverable: D3.4 Solutions for Data Aggregation, Cleaning, Harmonization & Storage

Due date of deliverable: (30-06-2021)

Actual submission date: (21-12-2021)

Start date of Project: 01 November 2017

Duration: 48 months

Responsible WP: FORTH

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 777167		
Dissemination level		
PU	Public	x
PP	Restricted to other programme participants (including the Commission Service	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (excluding the Commission Services)	

0. Document Info

0.1. Author

Author	Company	E-mail
Konstantinos Perakis	SiLo	kperakis@singularlogic.eu
Gianna Tsakou	SiLo	gtsakou@singularlogic.eu
Haridimos Kondylakis	FORTH	kondylak@ics.forth.gr
Valia Kalokyri	FORTH	vkalokyri@gmail.com
Panagiotis Argyropaidas	FORTH	parg@ics.forth.gr
Eleni Kolokotroni	ICCS	ekolok@mail.ntua.gr
Georgios Stamatakos	ICCS	gestam@mail.ntua.gr
Robert Richter	Noona	Robert.Richter@varian.com

0.2. Documents history

Document version #	Date	Change
V0.1	01/05/2021	Starting version, template
V0.2	02/05/2021	Definition of ToC
V0.3	25/05/2021	First complete draft
V0.4	10/06/2021	Integrated version (send to WP members)
V0.5	15/06/2021	Updated version (send PCP)
V0.6	22/06/2021	Updated version (send to project internal reviewers)
Sign off	29/06/2021	Signed off version (for approval to PMT members)
V1.0	30/06/2021	Approved Version to be submitted to EU
V1.1	16/12/2021	Revised Version based on reviewers comments

0.3. Document data

Keywords	Data aggregation, Data Cleaning, Harmonization
Editor Address data	Name: Konstantinos Perakis Partner: SiLo Address: Phone: Fax: - E-mail: kperakis@ep.singularlogic.eu
Delivery date	16/12/2021

1. Table of Contents

0. Document Info	2
0.1. Author	2
0.2. Documents history	2
0.3. Document data	2
1. Table of Contents	3
2. Introduction	5
3. BOUNCE Data Aggregation	7
3.1. Noona Core application	7
3.2. Noona eCRF application (Noona Studies)	8
3.3. Data quality controls	10
3.4. Data export process	12
4. BOUNCE Data Cleaning	14
4.1. Automated data cleaning	14
4.2. Enhanced data cleaning	23
5. BOUNCE Data Harmonization & Storage	28
5.1. Semantic Data Mapping Overview	30
6. Conclusions	34
7. References	36

Table of Figures

Figure 1: BOUNCE High level data flow	6
Figure 2: Example of a data collection form in the Noona Core application	7
Figure 3: Example of a summary of a filled form in the Noona Core application	8
Figure 4: An example of a data collection form in the Noona eCRF application	9
Figure 5: An example of the summary of a filled form in the Noona eCRF application	9
Figure 6: The data collection timeline of the Bounce study to guide the data-collection	10
Figure 7: Noona overall quality check for errors before submission	11
Figure 8: Example of a question in a Noona form with limited answer options.	11
Figure 9: Example of a question in a Noona form with an input fields accepting numeric values with a specific range	11
Figure 10: Example of a quality control in a Noona form with date input	12
Figure 11: BOUNCE data cleaning process	16
Figure 12: BOUNCE Data Cleaner high-level architecture	18
Figure 13: BOUNCE Data Cleaner – validation rules definition	18
Figure 14: BOUNCE Data Cleaner – cleaning rules definition	19
Figure 15: BOUNCE Data Cleaner – missing value handling rules definition	19
Figure 16: BOUNCE Data Cleaner – Records for verification	19
Figure 17: Example 1 of applied rules	21
Figure 18: Example 2 of applied rules	22
Figure 19: Example 3 of applied rules	22
Figure 20: Example of applied rules on external trial data provided by IEO	23
Figure 21: BOUNCE enhanced data cleaning in the holistic approach	23
Figure 22: Example of quality checks: Distribution of neutrophils per clinical site and the association between neutrophils and leukocytes per clinical site	25
Figure 23: Example of quality checks: Scatterplots depicting the differences in categorical variables codings used	25
Figure 24: Example of analysis of missingness for HUS subset	26
Figure 25: Data harmonization & storage process	28
Figure 26: An overview of the BOUNCE dataset available in the BOUNCE data repository	29
Figure 27: An overview of the IEO external dataset available in the BOUNCE data repository ..	30
Figure 28: D2RQ Architecture	31
Figure 29: Structure of an example D2RQ map	32
Figure 30: RDF Graph example about the TIPI questionnaire	33

2. Introduction

The scope of the deliverable D3.4 entitled “Solutions for Data Aggregation, Cleaning, Harmonization & Storage” is to undertake the documentation of the efforts carried out within the context of Task 3.3 - Data Source Aggregation & Cleaning and Task 3.4 - Data Source Harmonization & Storage. In this context, the deliverable D3.4 documents the methods and workflows that are designed and delivered in order to produce the aggregated, cleaned, harmonized and integrated datasets that have been made interoperable based on the BOUNCE semantic model. The produced interoperable datasets are leveraged in the data analysis that is performed within the context of the BOUNCE platform.

To this end, the purpose of the deliverable at hand can be summarized as follows:

- To document the data collection and aggregation process that is followed within the context of BOUNCE. The delivered process has the Noona ecosystem as the core element. At first, the two complementary applications, namely the Noona Core and the Noona eCRF applications, which are leveraged by both the patients and the clinical staff under different cases, are presented. Following the presentation of the applications, a detail description of the strict data quality controls which are employed on these applications are presented by documenting how all the input data are validated and verified before they are introduced in the BOUNCE platform. Finally, the provided data export process is documented in detail.
- To present the holistic data cleaning approach that is adopted in BOUNCE. At first, the automatic cleaning process is presented in detail. The design specifications of the automatic cleaning process are documented, followed by the implementation details of the BOUNCE Data Cleaner which implements this automatic cleaning process. In addition to this, the data validation, data cleaning and data completion rules are presented along with a set of examples that illustrate how the complete process is leveraged for both aggregated data from the questionnaires of the clinical sites of BOUNCE and external data such as the external trial data provided by IEO. Secondly, the enhanced data cleaning process, that is complementing the automatic cleaning process, is presented in detail. The three distinct phases of enhanced data cleaning process, namely the screening, diagnosing and post-processing phases, are thoroughly documented along with a set examples that illustrate their execution.
- To document the data harmonization and storage process that is designed and implemented within the context of BOUNCE. The process is comprised of the aggregation of the data, both from the clinical sites of BOUNCE as well as from the external trial data provided by IEO, into a single database, their recoding, and finally their exposure through both a relational API and a semantic API. The deliverable presents the set of tools which are exploited towards the efficient and effective exposure of both relational and semantically annotated data.

Figure 1 illustrates the complete solution of BOUNCE for data aggregation, data cleaning, data harmonization and storage. It illustrates how the various methods are combined in order to prepare the data for the upcoming data analysis. At first, the data are collected and aggregated using the Noona system for HUS, IEO and CHAMP and Qualtrics/Excel for HUJI. Those data are anonymized and exported to the clinical centres, which upload them to the BOUNCE Data Lake. In the case of external data, the data are uploaded to the BOUNCE Data Lake in a similar manner. Within the Data Lake, data cleaning is performed and the data are staged in the data lake. In a next step the data from the individual centers or external data are homogenized, mapped to the

ontology and exposed through a relational API and an RDF API in order to be further exploited by the models.

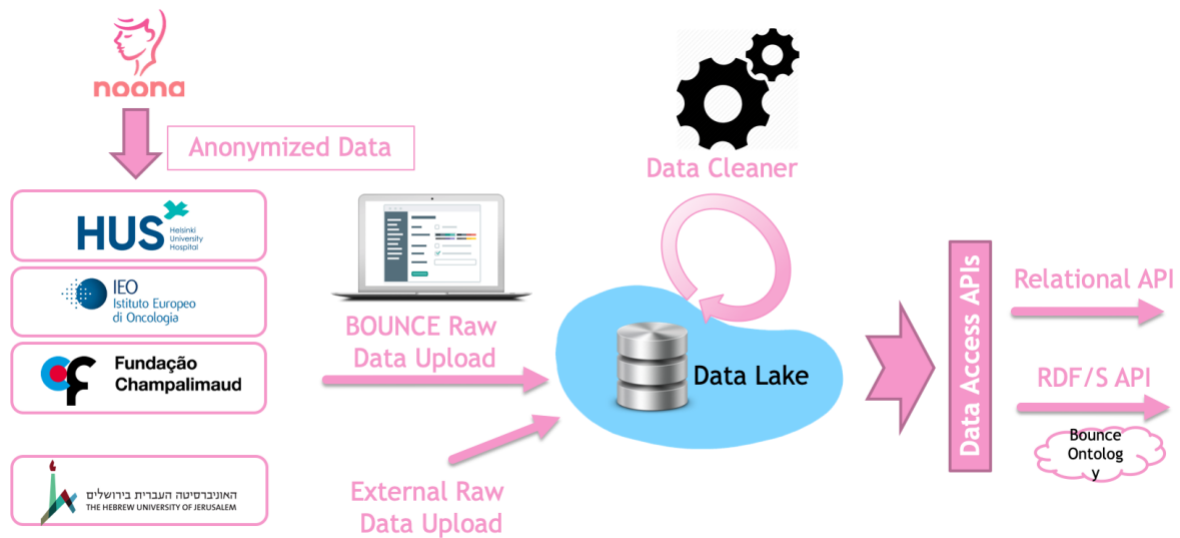


Figure 1: BOUNCE High level data flow

The document at hand is the revised version (v2.0) of the deliverable D3.4. The deliverable has been revised with the aim of documenting the updated information on the data quality controls which are applied on the Noona applications, as well as the data cleaning and harmonization operations performed on the data provided from the external (European Institute of Oncology (IEO) trial besides the data originating from the BOUNCE clinical sites. In addition to this, the updated document provides the details of the datasets which are available on the BOUNCE platform.

The current deliverable concludes all the activities performed in the specific work package (WP3), delivering the required data processing workflows which facilitate the efficient and effective integration of the collected datasets with the aim of maximizing their reuse, data quality and knowledge extraction generation.

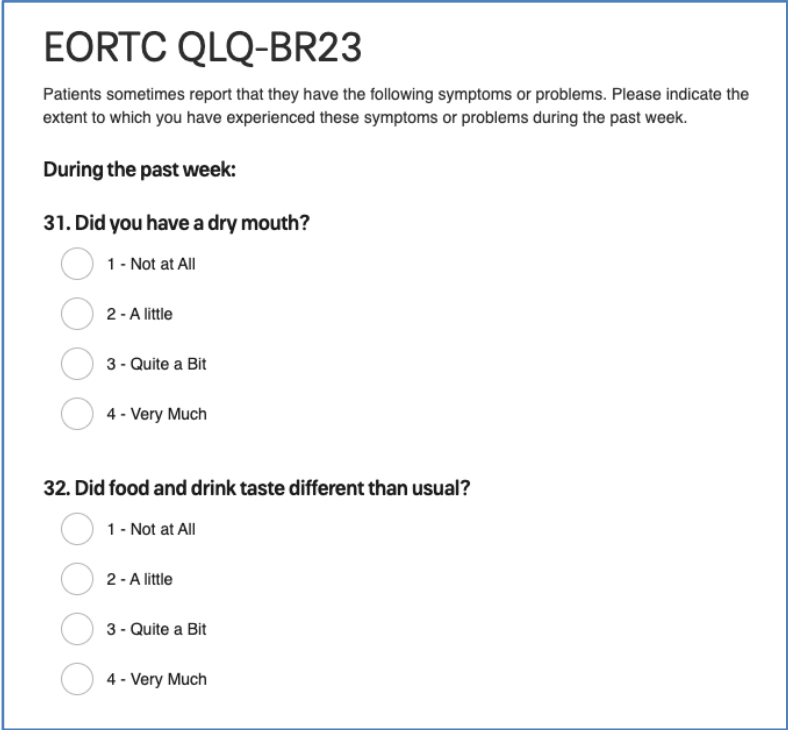
3. BOUNCE Data Aggregation

In the context of the BOUNCE project, data collection and aggregation is performed via the Noona system and it is realised via two complementary applications, namely the Noona Core and the Noona eCRF applications. On the one hand, the Noona Core application is used by patients and clinical staff during the daily treatment routine. On the other hand, the Noona eCRF application is used by the clinical staff to gather additional study-specific data that is not captured by the Noona Core application. Both applications provide export mechanisms which allow the extraction of data in a CSV file format.

3.1. Noona Core application

The Noona Core application consists of two parts, the patient application and the clinic application. The Noona patient application, which is offered as both a web and a mobile application, is used by patients to communicate with their clinics, as well as to answer various forms related to their treatment and general health status. Patients can fill such forms pro-actively (e.g., to report a new symptom to the clinic) or reactively in case the clinic has sent a questionnaire to the patient (e.g., a regular screening or a check-in before the next treatment visit).

On the other hand, the Noona clinic application is a web application that is used by the clinical staff in daily treatment tasks, such as the communication with patients, the organization of treatment visits and the sending of questionnaires. Furthermore, a nurse or a doctor can fill in the forms on behalf of the patient via the Noona clinic application, in case the patient is not able or willing to use the Noona patient application.



EORTC QLQ-BR23

Patients sometimes report that they have the following symptoms or problems. Please indicate the extent to which you have experienced these symptoms or problems during the past week.

During the past week:

31. Did you have a dry mouth?

☐ 1 - Not at All

☐ 2 - A little

☐ 3 - Quite a Bit

☐ 4 - Very Much

32. Did food and drink taste different than usual?

☐ 1 - Not at All

☐ 2 - A little

☐ 3 - Quite a Bit

☐ 4 - Very Much

Figure 2: Example of a data collection form in the Noona Core application

In both applications, questions are rendered with various standard input components like checkboxes, radio lists and free text input fields. The collected data elements will be identical independent of whether the patient filled in the form him/herself or whether a nurse or a doctor

filled in the form. Nevertheless, the source (patient vs. clinic staff) is clearly documented in the dataset for tracking and consistency purposes.

Sociodemographic information M3

Date answered
25.06.2021

5. What is your employment status?
Employed full time

7. If you are employed or self-employed, how many sick leave days did you have in the last three months?
1

8. If you are employed, have you had support from your employer to better/more flexible work arrangements?
Yes

19. a) During the last three months have you seen a mental health professional inside and/or outside of the hospital?
No

b) If yes, how many times?
2

20. Has your partner or someone in your family reduced their work time to take care of you during the last three months?
No

Yes. Please specify how many hours, days, months:
20 / week

21. Do you do any activities to support your well-being?
Yes

Yes. Please specify:
Sport

22. Have you used any services to support your well-being during the last three months?
No

23. Have you had any domestic help during the last three months? How many days?
Yes

Yes. Please specify number of days:
5

Figure 3: Example of a summary of a filled form in the Noona Core application

3.2. Noona eCRF application (Noona Studies)

The Noona eCRF application is used for the clinical data collection that involves manual data collection from the EMR or other non-integrated systems. Hence, the scope of this application is to collect complementary study-specific data based on the needs of the specific clinical study. Within the context of BOUNCE, different data collection forms have been implemented specifically for the purposes of the project and are filled by clinical staff. In particular, the forms include a variety of different measurements based on the input received from the outcomes of the worked performed in WP1 and WP6, and specifically deliverables D1.3 “BOUNCE methodology” [1] and D6.1 “Clinical pilot methodology and preparatory actions” [2].

Treatment update and side effects

Please fill out this form with data and measurements that happened between 04/07/2019 and 03/01/2020.

T6 End date for Anti HER2

Patient started a Anti HER2 treatment with Trastuzumab on 31/12/1999.

☒ End date

31/12/1999

☐ Patient is still on this therapy

T7a Side effects: Osteoporosis

☐ Yes, diagnosed on:

DD/MM/YYYY

☐ Yes, but date of diagnosis is unknown

☒ No

☐ Not documented / Unknown

T7b Side effects: Cardiac failure types

Asymptomatic decrease of LVEF	<input type="radio"/> Yes	<input type="radio"/> No	<input checked="" type="radio"/> Not documented / Unknown
Heart failure	<input type="radio"/> Yes	<input type="radio"/> No	<input checked="" type="radio"/> Not documented / Unknown
Coronary sdr	<input type="radio"/> Yes	<input type="radio"/> No	<input checked="" type="radio"/> Not documented / Unknown

Figure 4: An example of a data collection form in the Noona eCRF application

Treatment update and side effects

SUBMITTED Last updated on 01/06/2021 11:44 by Markus Manager

T6	End date for Anti HER2	End date: Fri Dec 31 1999
T7a	Side effects: Osteoporosis	No
T7b	Side effects: Cardiac failure types	
	• Asymptomatic decrease of LVEF	Not documented / Unknown
	• Heart failure	Not documented / Unknown
	• Coronary sdr	Not documented / Unknown
T7c	Side effects: Neutropenic infection	No
T7d	Other side effects	Patient didn't experience any other side effect
T7e	What other side effects did the patient experience	

Figure 5: An example of the summary of a filled form in the Noona eCRF application

In addition to the forms, the Noona eCRF application also displays a timeline according to the defined BOUNCE study protocol that highlights the different data collection points.

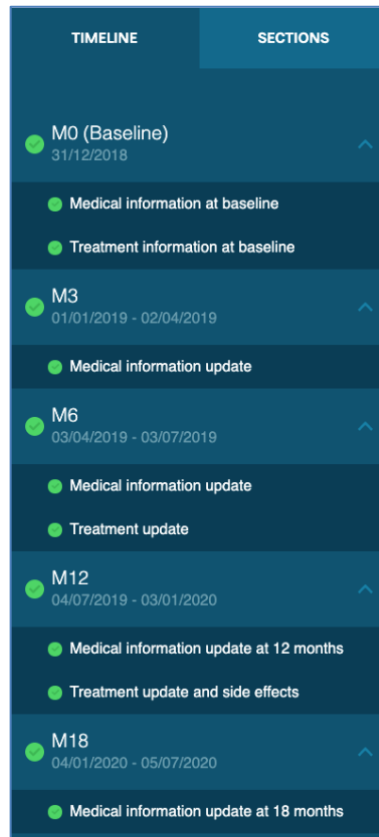


Figure 6: The data collection timeline of the Bounce study to guide the data-collection

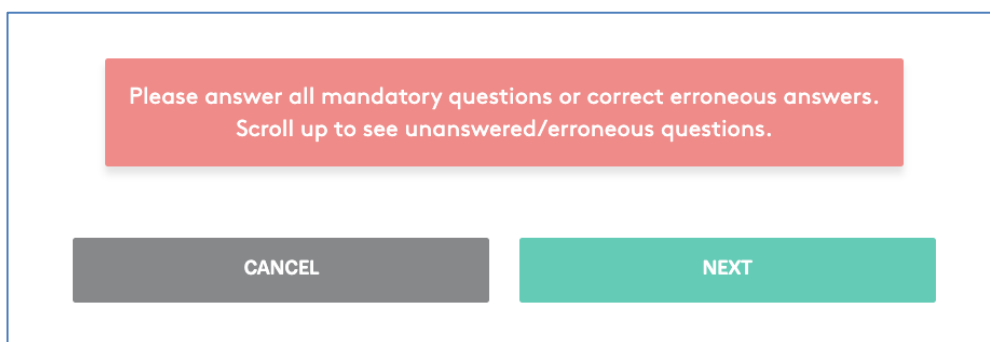
3.3. Data quality controls

During data entry and to ensure the highest quality of the collected data, a set of validation rules and data values controls are implemented in both the forms in the Noona Core application as well as the forms in the Noona eCRF application.

To achieve this, the designed validation rules are tailored to the corresponding forms and enforce checks such as the existence of answers in the mandatory fields and the conformance of values with regards to the defined valid value ranges for all input fields.

Strict engineering processes and best practices of software engineering including extensive unit and manual testing as well as mandatory code reviews ensure that forms are implemented according to the specifications and the data quality controls function correctly. In addition to this, heavy focus on usability and user centric design ensures that Noona forms are well usable by the different user groups.

Every identified error due to a rule violation will automatically trigger a clear and informative error message which is displayed to the user through the user interface of the applications. Unless all deployed rules are satisfied, the applications do not allow the submissions of the form to the system. Hence, If an input rule is violated, the form cannot be submitted and if the user tries to proceed by pressing the NEXT button, an error message will indicate that pending rule violations need to be handled (see Figure 7).

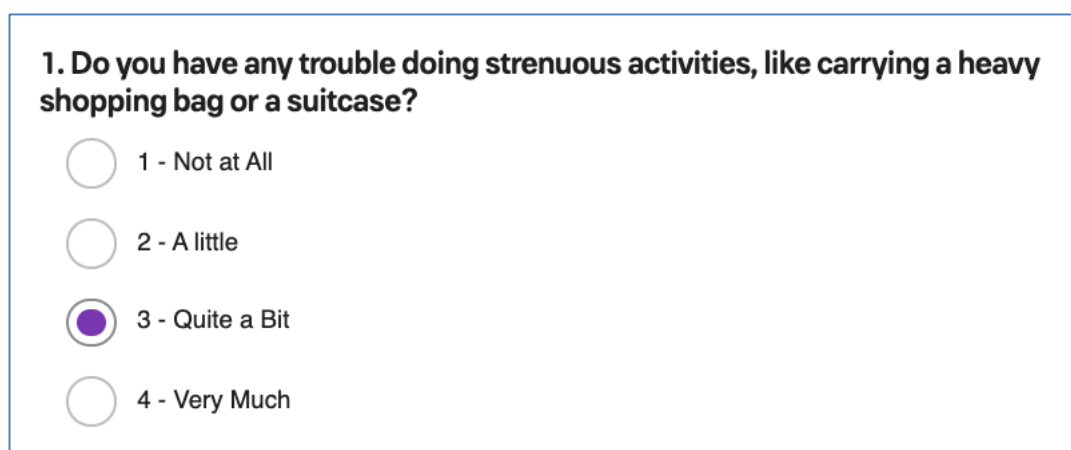


Please answer all mandatory questions or correct erroneous answers.
Scroll up to see unanswered/erroneous questions.

CANCEL NEXT

Figure 7: Noona overall quality check for errors before submission

Data entry forms in Noona rely heavily on pre-defined (multiple or singular choice) answers. This assures that data points are in a plausible range and can be reliably interpreted. An example of question in a form is depicted in Figure 8. As illustrated, only one option can be selected as an answer and all possible answers are clearly defined.



1. Do you have any trouble doing strenuous activities, like carrying a heavy shopping bag or a suitcase?

☐ 1 - Not at All

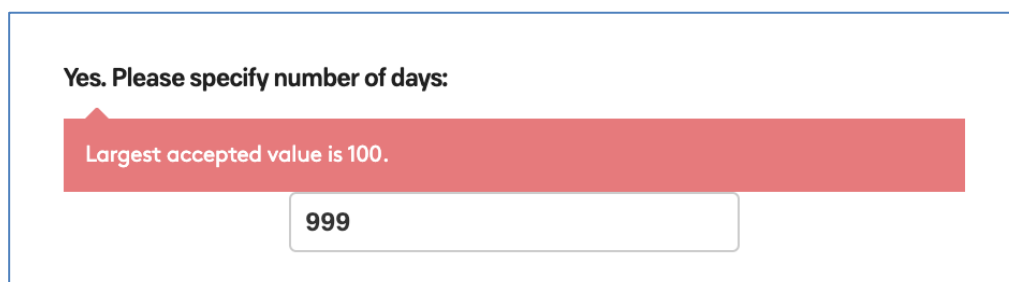
☐ 2 - A little

☒ 3 - Quite a Bit

☐ 4 - Very Much

Figure 8: Example of a question in a Noona form with limited answer options.

In the case of numeric values, input fields are utilized that suppress characters which are not numeric. Hence, it is only possible to enter valid values and e.g., not to provide as input the word “thousand” instead of entering the number 1000. In principal, numeric values are gathered via specialized input elements that do not allow to enter non number characters (except digits for floating point numbers if applicable). Moreover, based on the form specifications, as instructed by the BOUNCE consortium, range controls are also implemented in numeric values that limit the potential input even further besides safeguarding the provided input quality.



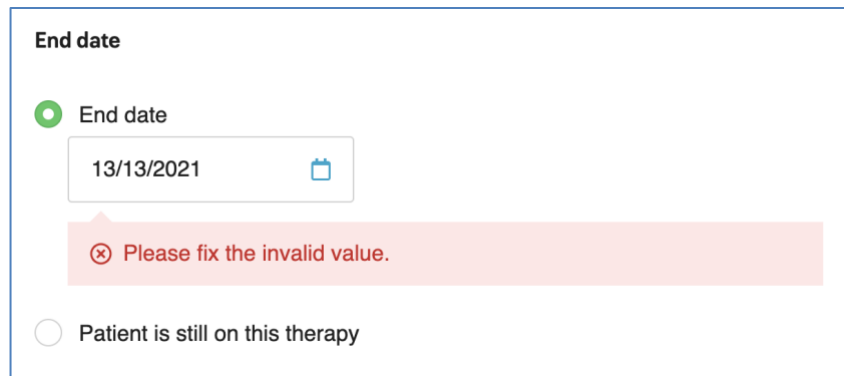
Yes. Please specify number of days:

Largest accepted value is 100.

999

Figure 9: Example of a question in a Noona form with an input field accepting numeric values with a specific range

Dates are collected also via specialized input elements that ensure consistent date formats and valid values. In this case, the date input element enforces a consistent format and prevents invalid values from being entered.



End date

☒ End date

13/13/2021

Please fix the invalid value.

☐ Patient is still on this therapy

Figure 10: Example of a quality control in a Noona form with date input

The forms in the Noona core application and the forms in the Noona eCRF (clinical data collection) application follow the same general structure and utilize the same base input components as explained above.

All gathered form data is stored in a relational data model in a state-of-the-art relational database. The table structure follows the structure of the visual forms in the user interface. Hence, table columns represent questions, and the cells contain the answers. Relational databases guarantee consistent and sound data by implementing transactions with ACID (Atomicity, Consistency, Reliability, Durability) semantics. This means, that data is either saved fully successfully or not at all. In case of a technical error during a save operation, the user interface would notify the user and the Noona application backend logging system would track the problem. Backend logs are scanned regularly for technical errors as part of Noona's operational procedures. Relational databases utilize strict schemas for the tables that store the data. E.g. schemas define what type of data elements can be put into which table columns. They provide an additional safety layer that limits the risk of values being stored on wrong columns, and hence, getting mixed up. The latter is also effectively prevented at development time by usage of a strongly typed programming languages for the backend code. Noona backend is implemented with such languages (Java, Kotlin). Such languages assignee strict types to variables (e.g., this variable represents an answer to question X and can assume the values A, B and C) in the program code and prevent the developer from implement wrong mapping code (e.g., putting the answer to question X on the field that's supposed to store the answer to question Z).

In summary, due to the controls described above data gathered via the Noona application are of the highest quality with respect to the specifications in the study protocol.

3.4. Data export process

The collected data can be easily exported from both the Noona Core application as well as from the Noona eCRF application in the form of CSV files. The data export process in both applications is as follows:

- Each clinical site has appointed person(s) from the clinical staff who are permitted to request data exports.
- The clinical staff member is requesting a data export via the Noona clinic application or Noona eCRF application respectively using the data export request feature.

- Upon receiving the data export request, a Varian employee with administrative rights approves the export via the same application.
- Upon this approval, the clinical staff member can download the exported data via the application user interface.

In the described export process several filters can be used to limit the data to specific forms or patient cohorts. In the Noona eCRF application, each study site needs to export their data individually and the data file will only contain data of the corresponding site.

The files are always exported in a password protected zip file. The password is sent to the requesting clinical staff member via SMS. In the Noona clinic application, the SMS with the password is sent only when the user tries to download the export file. In the case of the Noona eCRF application, the password is sent when the export request has been approved by the Varian administrator. In both applications, the export files are only available for 24 hours after approval. The produced export files are not stored on external systems. All data remains protected and encrypted in the corresponding database until they are downloaded by the user.

The data is read directly from the relation database with a read-only operation that will not modify the data. The code that implements the export mechanism is implemented based on the same high quality engineering procedures already described in section 3.3. These ensure that data is correctly extracted from the database tables and mapped into the CSV structure. Noona form data is not modified by any process after it has been saved. Hence, the exported data can be assumed to be of high quality due to the controls described in 3.3.

4. BOUNCE Data Cleaning

The data cleaning process aims at detecting and correcting (or removing) any “messy”, “noisy”, corrupted or erroneous data entries of a dataset. To this end, the scope of the process is to provide the means to increase the data quality of a dataset, which is usually composed of information originating from a variety of heterogeneous data sources, by increasing the cleanliness and completeness of the dataset to the highest possible degree. It involves multiple phases during which the various incomplete, incorrect or corrupted data entries of a dataset are identified and corrected, transformed or discarded from the dataset.

Within the context of BOUNCE, the consortium adopted a holistic approach for the data cleaning operations which are performed on the prospective data that are collected from the clinical sites, as well as the data from the external (European Institute of Oncology (IEO) trial. At first, the incoming prospective data or data from the external trial are fed to the automated cleaning process of BOUNCE in order to be “cleaned” and stored in BOUNCE Data Lake in an automated and robust manner. In the second and final step of the data cleaning approach, the “cleaned” data are fed again to a further enhanced data cleaning process which involves manual intervention and close collaboration with the clinical partners. In the following paragraphs, the holistic data cleaning approach of BOUNCE is presented in detail.

4.1. *Automated data cleaning*

Data cleaning is a complex process in which multiple aspects of the underlying data should be taken into consideration in order to design and execute effective and efficient data cleaning operations. In all data collection processes, the data entries of a dataset are typically acquired in large volume with great diversity and from different heterogeneous data sources. As a consequence, the data collection usually introduces data entries which contain errors, as well as dirty and coarse information. Typically, this includes malformed data entries, typos, missing values, duplicate values and outliers.

Hence, it is imperative that a data cleaning process is established in order to provide the required data cleaning operations on top of the collected datasets. The purpose of this process is to clean and complete the acquired information towards the increase of its quality, completeness, correctness and value.

Within the context of BOUNCE, an automated data cleaning process has been designed whose main characteristics can be grouped in the following two axes:

- A flexible data validation mechanism built around a set of rules which are capable of identifying the underlying erroneous data entries. This mechanism is effectively covering the complete spectrum of the data quality dimensions and is embracing the state-of-art data validation practises.
- A complete data cleaning workflow execution mechanism which is empowered with the most efficient and effective data cleaning and data completion techniques that are capable of correcting or eliminating the identified errors and problematic data entries.

The designed process is composed of a set of sequential steps which span from the preliminary analysis of the underlying data, the definition of the appropriate data validation and the composition of the data cleaning workflow, to the assessment and verification of the produced “clean” dataset. To this end, the sequential steps of the BOUNCE data cleaning process are as follows:

- **Data analysis of the dataset:** The scope of this step is to analyse and identify the data included in the underlying dataset. This initial step is very crucial as the results of this

analysis will drive the design of the validation rules and the whole data cleaning workflow. Through this analysis, the different data fields of the datasets are identified and for each data field the corresponding data aspects are collected. The data aspects include, among others, the type of data, the acceptable format of the data entries, the list of acceptable (in terms of either range or distinct) values, as well as any limitations or restrictions.

- **Data validation of the dataset:** The scope of this step is to define the appropriate rules to which the data entries of the underlying dataset should conform. The defined rules will be the basis of the data cleaning workflow since they will be utilised so as to detect the inconsistencies, erroneous entries and any missing data entries in the dataset. The list of errors or inconsistencies will be fed to the data cleaning and missing value handling mechanism in order to be addressed or eliminated. The definition of these rules requires deep knowledge of the respective dataset as well as domain knowledge of the information included in the dataset. Hence, the input for these rules is collected through a close collaboration of the technical partners and the clinical partners. Furthermore, this step ensures the high level of customisation of the data cleaning process since different datasets can have different sets of validation rules.
- **Design of the data cleaning workflow:** The scope of this step is to define the appropriate data cleaning and missing value handling rules that will be applied in order to correct or eliminate the conformance errors identified by the validation rules. On the one hand, the data cleaning rules will either correct or remove the erroneous data entries based on the input received. On the other hand, the missing value handling (data imputation) rules will be applied in order to automatically fill-in the missing values in the identified problematic data entries of the dataset. Both the data cleaning and missing value handling rules are connected to a validation rule whose execution results will trigger the execution of the corrective actions on the non-conforming data entries.
- **Execution and Verification:** The scope of this step is to execute the designed data cleansing workflow which is composed by the set of validation rules for the data fields of the dataset which are bound to the corresponding data cleaning and missing value handling rules. Hence, the result of this execution will be the “cleaned” dataset that will now conform to all validation rules. This final step includes also the verification of the performed operations with the inspection of the detailed records that contain the details of the identified errors and the corrective actions that were undertaken in order to correct them or eliminate them.

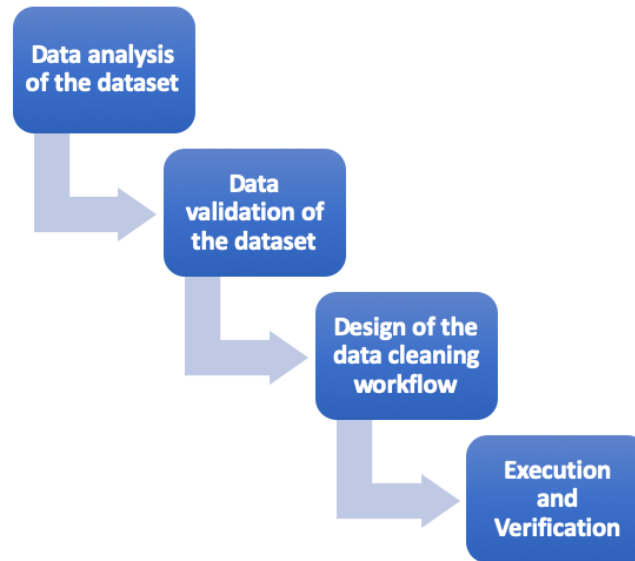


Figure 11: BOUNCE data cleaning process

The realisation of the presented BOUNCE data cleaning process is performed by the BOUNCE Data Cleaner component which is integrated the BOUNCE platform. Following the specifications of the presented data cleaning process, the BOUNCE Data Cleaner comprises of five different sub-components, each one of which undertakes a specific step of the process.

The **Workflow Executor** sub-component constitutes the main sub-component of the BOUNCE Data Cleaner. It orchestrates the whole data cleaning workflow execution by utilising the rest of the sub-components and their respective services. The specific sub-component receives as input the data validation, data cleaning and missing value handling rules and the dataset that will be cleaned in order to execute the data cleaning workflow execution by invoking the services of the rest of the sub-components. It offers the single user friendly and easy-to-use user interface of the BOUNCE Data Cleaner through which the user is able to select and load the selected dataset and perform the preliminary analysis of the dataset. In addition to this, this sub-component offers the ability to define all the data validation, data cleaning and missing value handling rules and trigger the data cleaning workflow's execution through its user interface. Finally, once the execution is finished, it displays the detailed records which illustrate the identified errors and the corrective actions that were executed for verification purposes. Besides the user interface, this sub-component is offering the possibility to execute the process as a fully automated background process through the RESTful API that it offers. In detail, the sub-component offers the corresponding endpoint which can be utilised in order to trigger the execution of the data cleaning workflow by providing the required information, such as the rules that will be applied as well as the dataset on which these rules will be applied.

The **Validator** sub-component is executing the data validation of the selected data entries of the input dataset. Hence, it provides the implementation of all the offered data validation checks of the BOUNCE Data Cleaner. The data validation rules are translated into a set of constraints that all the data entries of a selected data field should conform with. In the case of a non-conforming data entry, the specific data entry is marked as an erroneous entry that is handled by the corresponding data cleaning or missing value handling rule. The list of data validation rules includes the following:

- Conformance to a specific data type (i.e., boolean, integer, string, etc.).

- Conformance to a pre-defined value range (i.e., the minimum and maximum acceptable values).
- Conformance to a list of pre-defined values (i.e., “Yes” or “No”, or [1,2,3]).
- Conformance to value representation (i.e., all dates are following the “yyyymmdd” format).
- Conformance to regular expression patterns (i.e., data entries should adhere a specific pattern).
- Conformance to value uniformity (i.e., all time-stamps are in UTC format).
- Conformance to uniqueness (i.e., duplicate values are not acceptable).
- Conformance to non-empty value (i.e., all mandatory fields should have values).
- Conformance to cross-field validity (i.e., the sum of fields with percentage values must be equal to 100).
- Conformance to cross-field dependency (i.e., in case the field is set to a value then the other field should be set to a value).

The **Cleaner** sub-component is performing the cleaning operations on the selected data entries of the input dataset. In particular, it executes the appropriate cleaning actions in order to correct or eliminate the non-conforming data entries from the datasets, as identified by the Validator. The data correction actions which are executed are based on the defined data cleaning rules. In detail, the offered data correction actions are as follows:

- The rejection of an inconsistent value and the removal of the specific value from the data fields of the dataset.
- The rejection of an inconsistent value and the removal of the complete record or line from the dataset.
- The replacement of an inconsistent value with a statistical value, such as the minimum or maximum value observed, the mean or median value or the most frequent value.
- The replacement of an inconsistent value with a pre-defined value as set in the data cleaning rule.

The **Completer** sub-component is responsible for ensuring the completeness of the data fields of the dataset. In particular, it safeguards the existence of data entries in the mandatory data fields by checking the conformance of the specific data fields to the defined data validation rules. In the case of non-conforming data entries, it utilises several missing value techniques in order to perform automatic filling of the missing values depending on the requirements set on the data completion rules, as well as the nature of each specific data field. In detail, the offered missing value handling techniques include the following:

- The utilisation of the most common statistical methods, such as the mean or median value, the minimum or maximum observed value or the most frequent value.
- The utilisation of the Linear Regression algorithm.
- The utilisation of the k-Nearest Neighbours algorithm.
- The utilisation of the Moving Average method.
- The utilisation of the Last Observation Carried Forward (LOCF) and Next Observation Carried Backward (NOCB) methods.
- The automatic fill-in of missing values with a predefined value.

The **Recorder** sub-component undertakes the recording of all the corrective actions which are performed during the execution of the data cleaning workflow. In particular, it generates

detailed records for each action that each sub-component performs during the execution of the process. The records contain the information related to the error identified due to the lack of conformance on a specific validation rule, as well as the data cleaning or data completion action that was performed to eliminate the error. The detailed records are also provided for inspection and verification at the end of the process.

Figure 12 depicts the high-level architecture of the BOUNCE Data Cleaner.

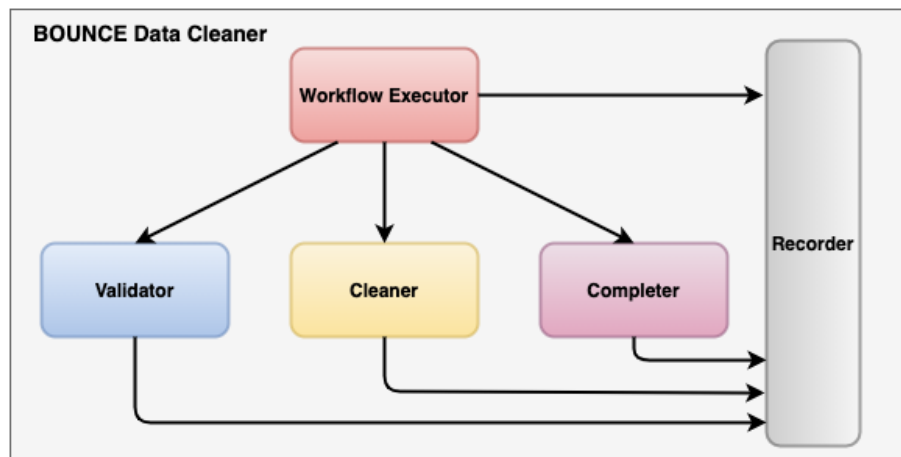


Figure 12: BOUNCE Data Cleaner high-level architecture

The following figures illustrate the basic aspects of the user interface of the BOUNCE Data Cleaner.

The screenshot shows the 'Validation Rules' section of the BOUNCE Data Cleaner interface. It includes a navigation bar with links for 'Validation Rules', 'Datasets', 'Rules', 'Logs', and 'Clean'. A 'Sign out' button is in the top right. Below the navigation bar is a 'Back to Datasets' link. The main area displays a table of validation rules with columns for Variable, Validation Rule Name, Validation Rule Method, Extra Args, and Actions. The table contains four entries for '1.Yearofbirth' and 'Age'. At the bottom, there are buttons for 'Add New Validation Rule', 'Edit Cleaning Rules', and 'Edit Missing Value Handling Rules'. A pagination bar at the bottom indicates 'Showing 1 to 4 of 4 entries' and includes 'Previous', '1', and 'Next' buttons.

Variable	Validation Rule Name	Validation Rule Method	Extra Args	Actions
1.Yearofbirth	data_type_validation	DATA_TYPE	NONE	<input checked="" type="checkbox"/> <input type="checkbox"/>
1.Yearofbirth	birth_range	RANGE	1900, 2020	<input checked="" type="checkbox"/> <input type="checkbox"/>
Age	data_type_validation	DATA_TYPE	NONE	<input checked="" type="checkbox"/> <input type="checkbox"/>
Age	age_range	RANGE	0, 110	<input checked="" type="checkbox"/> <input type="checkbox"/>

Figure 13: BOUNCE Data Cleaner – validation rules definition

BOUNCE Data Cleaner Cleaning Rules					Sign out
Back to Validation Rules					
Show 10 entries	Search:				
Variable	Validation Rule Name	Cleaning Rule Method	Extra Args	Actions	
1.Yearofbirth	data_type_validation	NONE	NONE	✕	
1.Yearofbirth	birth_range	REPLACE_WITH_VALUE	-998	✕	
Age	data_type_validation	NONE	NONE	✕	
Age	age_range	REPLACE_WITH_VALUE	-998	✕	
Showing 1 to 4 of 4 entries					Previous 1 Next

Figure 14: BOUNCE Data Cleaner – cleaning rules definition

BOUNCE Data Cleaner Missing Value handling Rules					Sign out
Back to Validation Rules					
Show 10 entries	Search:				
Variable	Missing Value Handling Method	Extra Args	Actions		
1.Yearofbirth	FILL_WITH_VALUE	-999	✕		
Age	FILL_WITH_VALUE	-999	✕		
Showing 1 to 2 of 2 entries					Previous 1 Next

Figure 15: BOUNCE Data Cleaner – missing value handling rules definition

BOUNCE Data Cleaner Dashboard for file hus_mos_20210622_160350.log					Sign out
Cleaning Actions					
Show 10 entries	Search:				
Violated Constraint	Correction Action	Variable	Row Number		
REGEX_PATTERN_PASSTHROUGH	REPLACE_WITH_REGEX	Generallyspeaking,howoftenduringthepast4weekswereyouabletodowhatyourmedicalteamtoldyou?	71		
REGEX_PATTERN_PASSTHROUGH	REPLACE_WITH_REGEX	Generallyspeaking,howoftenduringthepast4weekswereyouabletodowhatyourmedicalteamtoldyou?	72		
REGEX_PATTERN_PASSTHROUGH	REPLACE_WITH_REGEX	Generallyspeaking,howoftenduringthepast4weekswereyouabletodowhatyourmedicalteamtoldyou?	73		
REGEX_PATTERN_PASSTHROUGH	REPLACE_WITH_REGEX	Generallyspeaking,howoftenduringthepast4weekswereyouabletodowhatyourmedicalteamtoldyou?	74		
REGEX_PATTERN_PASSTHROUGH	REPLACE_WITH_REGEX	Generallyspeaking,howoftenduringthepast4weekswereyouabletodowhatyourmedicalteamtoldyou?	75		
REGEX_PATTERN_PASSTHROUGH	REPLACE_WITH_REGEX	Generallyspeaking,howoftenduringthepast4weekswereyouabletodowhatyourmedicalteamtoldyou?	76		
REGEX_PATTERN_PASSTHROUGH	REPLACE_WITH_REGEX	Generallyspeaking,howoftenduringthepast4weekswereyouabletodowhatyourmedicalteamtoldyou?	77		
REGEX_PATTERN_PASSTHROUGH	REPLACE_WITH_REGEX	Generallyspeaking,howoftenduringthepast4weekswereyouabletodowhatyourmedicalteamtoldyou?	78		
REGEX_PATTERN_PASSTHROUGH	REPLACE_WITH_REGEX	Generallyspeaking,howoftenduringthepast4weekswereyouabletodowhatyourmedicalteamtoldyou?	79		
REGEX_PATTERN_PASSTHROUGH	REPLACE_WITH_REGEX	Generallyspeaking,howoftenduringthepast4weekswereyouabletodowhatyourmedicalteamtoldyou?	80		
Showing 181 to 190 of 224 entries					Previous 1 ... 18 19 20 ... 23 Next
Download log file					

Figure 16: BOUNCE Data Cleaner – Records for verification

As documented in Section 3, the data collection and aggregation from clinical sites is performed via two complementary applications from Noona. The collected and aggregated data are then exported and uploaded in the BOUNCE Data Lake for the cleaning process to be executed. In the same manner, in the case of any external data, such as the data from the external (European Institute of Oncology (IEO) trial, aggregated data are again uploaded in the BOUNCE Data Lake in order to be also cleaned.

The BOUNCE Data Cleaner is then engaged in order to perform the necessary corrective actions on each new dataset that is introduced in the Bounce Data Lake. For the definition of the data validation, data cleaning and missing value handling rules, the consortium was engaged into a

series of internal roundtable discussions and data analysis sessions where both the technical partners and the clinical partners participated.

In terms of validation, within the context of BOUNCE one suitable rule type was the “conformance to regular expression patterns” due to the nature of datasets which are based on the designed questionnaires, as described in Section 3. To this end, the rule named “REGEX_PASSTHROUGH” was inserted into the BOUNCE Data Cleaner in which the validity of the values of a specified column of a questionnaire was validated and the indices of the rows that do not follow the defined pattern were identified.

Another validation rule type that was embraced was the “conformance to a pre-defined value range”. The rule “RANGE” was inserted into the BOUNCE Data Cleaner in order to examine the validity of the values of a specified column against a list of comma separated numerical arguments. The indices of the rows that do not conform to the specified range are marked as errors. In some cases, the special arguments “INF/ -INF” were used in order to check the validity based on the lower and upper limit.

The third and final validation that was utilised was the “conformance to a specific data type”. The data type validation rules were inserted into the BOUNCE Data Cleaner when a new variable, represented by a column, was added in the data cleaning process. The list of validated data types includes the INTEGER, NUMERIC, DATE, STRING data types. The specific rules validate the values of a column against the declared type of the variable and it was utilised on all the collected data, both the ones originating from the questionnaires of the clinical centres of BOUNCE and the ones originating from the external trial provided by IEO.

In terms of data cleaning, three cleaning rule types were defined and were bounded to separate data validation rule instances. At first, the “REPLACE_WITH_REGEX” cleaning rule type was utilised that takes two arguments. The first argument is the regular expression pattern to interfere from the violated rows (indicated by the indices returned by the aforementioned validation rule). The second argument is the regular expression upon which cleaning will be based. The common pattern for this rule, regarding Bounce questionnaires are: **PatternA:** ‘-[a-zA-Z]*’ and **PatternB:** \0. In the specific rule, Pattern B takes the first regex group that satisfies PatternA. For example, if a value is in the form ‘5 – Strongly Agree’, the result of the application of the cleaning rule is number 5.

Secondly, the “REPLACE_WITH_MAPPING” cleaning rule type was utilised that takes always an even number of arguments. According to this rule, for every two arguments, the first one is the value of the dataset which is going to be replaced, while the second one is the replacing value. For example, given the arguments ‘Agree’, ‘1’, ‘Disagree’, ‘2’, ‘Not agree nor disagree’, ‘3’ the applied mapping to the dataset will have the below form:

Agree ---> 1

Disagree ---> 2

Not agree nor Disagree ---> 3

Finally, the third cleaning rule type that was utilised was the replacement of an inconsistent value with a pre-defined value. In this case, the non-conformant values of a column were replaced with the number -998 for the data from the questionnaires of the clinical centres of BOUNCE and with the number -1 for the data from the external trial that was provided by IEO. The presented rule types were applied across the different data fields of all the datasets that were collected through the various questionnaires of BOUNCE, as well as the utilised data from

the external trial as explained above. In total, 3137 rules were applied on the provided datasets, from which 2853 are related to the BOUNCE data and 284 are related to the external trial data. The following paragraphs present a list of examples of these rule types as they were applied on different datasets (from both the internal prospective and the external trial) within the context of the BOUNCE project.

For questionnaire ‘Sociodemographic information’ of provider ‘CHAMP’ and the columns ‘Age’ and ‘1.Year of Birth’, the BOUNCE Data Cleaner ensures that the values of both columns conform to a specific range. In this case, invalidated values are replaced with the number -998. Missing values are handled by automated fill-in with the number -999. Figure 17 depicts the aforementioned rules for column ‘Age’.

```
{
  "dataset": "sociodemographic",
  "missing_value_rule": {
    "method": "FILL_WITH_VALUE",
    "args": "-999"
  },
  "owner": "admin",
  "provider": "champ",
  "validation_rules": [
    {
      "name": "data_type_validation",
      "method": "DATA_TYPE",
      "args": "NONE",
      "cleaning_rule": {
        "method": "NONE",
        "args": "NONE"
      }
    },
    {
      "name": "age_range",
      "method": "RANGE",
      "args": "0,110",
      "cleaning_rule": {
        "method": "REPLACE_WITH_VALUE",
        "args": "-998"
      }
    }
  ],
  "variable": "Age"
}
```

Figure 17: Example 1 of applied rules

For questionnaire “Fare” of provider ‘HUS’ and column named “11. We have the strength to solve our problems”, the BOUNCE Data Cleaner ensures that all records are numeric. As far as validation service is concerned, the data type of every value of the column is checked based on regular expression pattern ‘^[0-9]*\$’. This pattern checks if the input string contains only integer numbers. Given that the values of the column were basically strings, the column was cleaned based on a specific mapping, while missing values are handled by automated fill-in with the number -999. Figure 18 depicts the aforementioned rules for column “11. We have the strength to solve our problems”.

```

{
  "dataset": "fare",
  "missing_value_rule": {
    "args": "-999",
    "method": "FILL_WITH_VALUE"
  },
  "owner": "admin",
  "provider": "hus",
  "validation_rules": [
    {
      "name": "apply_mapping",
      "method": "REGEX_PATTERN_PASSTHROUGH",
      "args": "^[0-9]*$",
      "cleaning_rule": {
        "method": "REPLACE_WITH_MAPPING",
        "args": "Strongly agree,7,
Moderately agree,6,
Slightly agree,5,
Nor disagree nor agree,4,
Slightly disagree,3,
Moderately disagree,2,
Strongly disagree,1"
      }
    }
  ],
  "variable": "11.We havethestrengthtosolveourproblems"
}

```

Figure 18: Example 2 of applied rules

For questionnaire 'TIP1' of provider 'IEO' and column '2.Critical,quarrelsome.', the BOUNCE Data Cleaner ensures that all values of the column are numeric. Given that the majority of the values were alphanumeric, they were validated based on the same regular expression pattern that was referenced on the previous example. With regards to the cleaning rule, the alphanumeric values followed the pattern `[0-9] - [a-zA-Z]*`. As a result of the application of the cleaning rule, the string part of the value was discarded, keeping only the number. Finally, missing values are handled by automated fill-in with the number -999, as with the previous examples.

```

{
  "dataset": "tipi",
  "missing_value_rule": {
    "args": "-999",
    "method": "FILL_WITH_VALUE"
  },
  "owner": "admin",
  "provider": "ieo",
  "validation_rules": [
    {
      "name": "data_type_validation",
      "method": "DATA_TYPE",
      "args": "NONE",
      "cleaning_rule": {
        "method": "NONE",
        "args": "NONE"
      }
    },
    {
      "name": "regex_pattern",
      "method": "REGEX_PATTERN_PASSTHROUGH",
      "args": "^[0-9]$",
      "cleaning_rule": {
        "method": "REPLACE_WITH_REGEX",
        "args": "' - [a-zA-Z ]*', '\\0'"
      }
    }
  ],
  "variable": "2.Critical,quarrelsome."
}

```

Figure 19: Example 3 of applied rules

For questionnaire 'IEO_EXTERNAL' of provider 'IEO' and column 'Genomic_test', the BOUNCE Data Cleaner first of all ensures that all values of the column are numeric. In addition to this, the BOUNCE Data Cleaner ensures that the values of the column are non-negative. In this case, invalidated values are replaced with the number -1. Finally, missing values are handled by

automated fill-in with the number -999, as with the previous examples. Figure 20 depicts the aforementioned rules for the column 'Genomic_test'.

```
{
  "dataset": "ieo_external",
  "missing_value_rule": {
    "method": "FILL_WITH_VALUE",
    "args": "-999"
  },
  "owner": "doctor",
  "provider": "ieo",
  "validation_rules": [
    {
      "name": "data_type_validation",
      "method": "DATA_TYPE",
      "args": "NONE",
      "cleaning_rule": {
        "method": "NONE",
        "args": "NONE"
      }
    },
    {
      "name": "",
      "method": "RANGE",
      "args": "0,INF",
      "cleaning_rule": {
        "method": "REPLACE_WITH_VALUE",
        "args": "-1"
      }
    }
  ]
},
"variable": "Genomic_test"
}
```

Figure 20: Example of applied rules on external trial data provided by IEO

4.2. Enhanced data cleaning

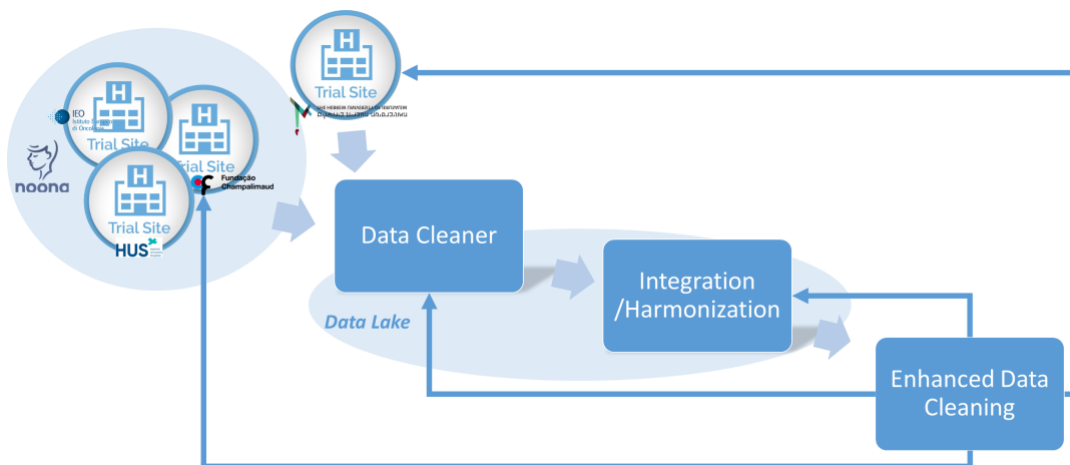


Figure 21: BOUNCE enhanced data cleaning in the holistic approach

Error prevention, diagnosis and treatment strategies adopted within BOUNCE (see paragraphs 3.3 and 4.1) can reduce many problems but cannot eliminate them all. A special problem is that of erroneous inliers, i.e., data points generated by error but falling within the expected range [3]. They often escape detection and are discovered only if viewed in relation to other variables [3] [4]. Furthermore, at clinical sites and BOUNCE Data Lake, prospective data undergo repeated steps of being entered into information carriers, extracted, transferred to other carriers, cleaned, edited, selected, transformed, summarized, integrated and presented. As Noona export files (CSV files) and HUJI files (Qualtrics/Excel files) differ in file format, naming conventions, categorical variable codings and columns, the need arises for them to be transformed into one

cohesive data set. Errors can occur at any stage of this data flow, such as corruption in file transmission or values incorrectly entered, changed, or transformed.

BOUNCE has adopted a holistic process of data cleaning, namely, the initially cleaned, homogenized and integrated dataset of BOUNCE prospective data has undergone a second round of cleaning process by BOUNCE researchers to identify any remaining faulty data or inconsistencies, to guarantee the completeness and efficiency of the implemented data cleaner workflow, to verify the correctness of harmonization/integration workflow and to ensure overall data quality (Figure 21). Enhanced data cleaning is taking place prior to statistical analysis and predictive modelling. It is repeated every time a new wave is integrated into the dataset. Three distinct phases of this data cleaning process can be identified: **screening**, **diagnosing** and **post-processing** of the suspected data abnormalities.

During **screening** phase, the data is audited to detect anomalies and contradictions. Screening methods have been relying on basic statistical methods, prior expectations and common sense. Data examination with simple descriptive tools is an essential first step to better understand the data. In particular, auditing steps performed include:

- **Detection of outliers:** Data are examined with basic descriptive tools. For every variable, records with values falling outside the expected range are identified.
- **Detection of inliers:** Comparison and consistency checks between variables, based on joint frequency distributions and scatterplots have been performed (Figures 22, 23). The aim is to identify erroneous inliers, i.e. faulty data with values falling within the expected range. Inliers could arise during data collection because of faulty value entry or during integration/harmonization because of faulty data conversion. Examples of consistency checks are the following: participants that smoke or drink wine are expected to have a non-zero value in number of cigarettes or glasses of wine respectively and vice versa, retired participants are expected to have a higher average age compared to employed participants, single women are expected to have less children than married or windowed, for neoadjuvant chemotherapy the chemo date should be before surgery date, etc.
- **Evaluation of differences between clinical sites** in variable distributions, either based on graphical means (e.g. scatterplots) or by using chi-square or Anova test (Figures 22, 23). The aim is to identify striking differences between HUII and the rest of the datasets, implying erroneous conversions during harmonization/integration of HUII datasets with Noona exports (used for HUS, IEO and CHAMP datasets).
- **Analysis of missingness:** Missing values may be due to interruptions or corruptions in the data flow or the unavailability of the target information (Figure 24). They require careful examination before treatment. Special attention is given to differences in missing patterns between the clinical sites for the reason mentioned above.
- **Recalculation of subscale scores** and comparison with the integrated dataset. The aim is to verify that questionnaire items are correctly mapped and converted to questionnaire subscales.
- **Checking the temporal validity of data.** Most Noona export files contain all participants' waves (time points) until the date of export. Assignment of each record to the correct wave is not always straight forward.
- **Dataset completeness:** Aims to examine whether all participants with records in the export files appear in the integrated dataset.

To this end, RapidMiner software tool, R programming language and packages and excel spreadsheets have been used. The following paragraphs document some examples of the auditing steps performed.

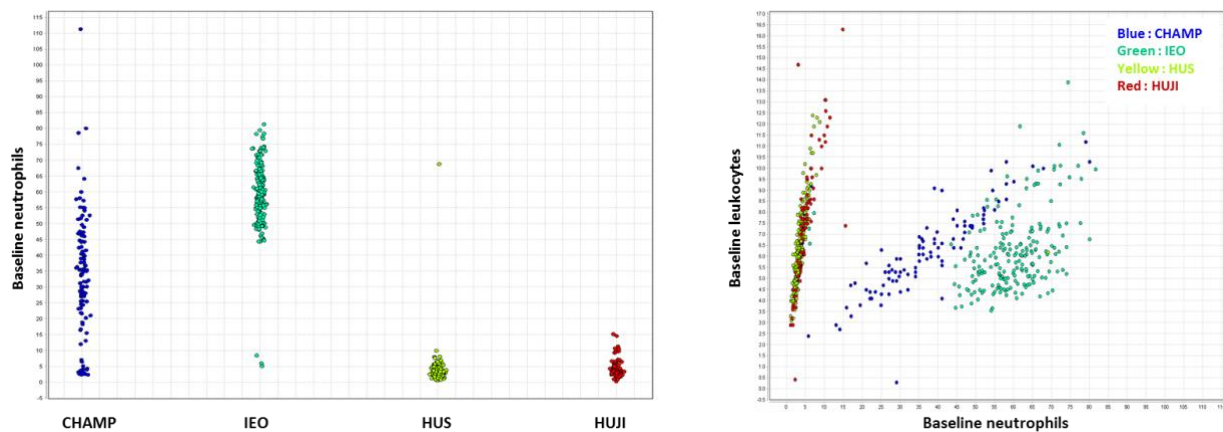


Figure 22: Example of quality checks: Distribution of neutrophils per clinical site and the association between neutrophils and leukocytes per clinical site

Figure 22 depicts the distribution of neutrophils per clinical site at baseline and the association between neutrophils and leukocytes per clinical site at baseline. Differences among clinical sites implied the need for an update of the unit conversion rules during harmonization/integration phase.

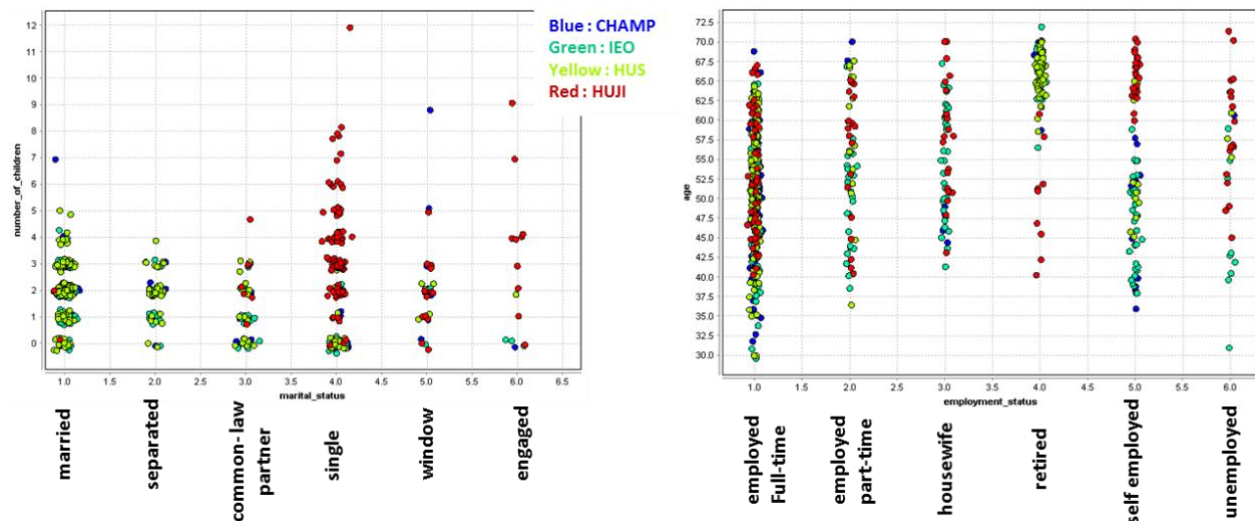


Figure 23: Example of quality checks: Scatterplots depicting the differences in categorical variables codings used

Figure 23 displays scatterplots between a) marital status and number of children and b) between employment status and age, per clinical site. In HUJI dataset the vast majority of participants appeared as single with children in contrast with other sites. Furthermore, in HUJI dataset retired participants are few and of younger age, whereas participants of older age appear mostly as self-employed. These strange patterns are the result of differences in categorical variables codings used in HUJI dataset compared to Noona exports.

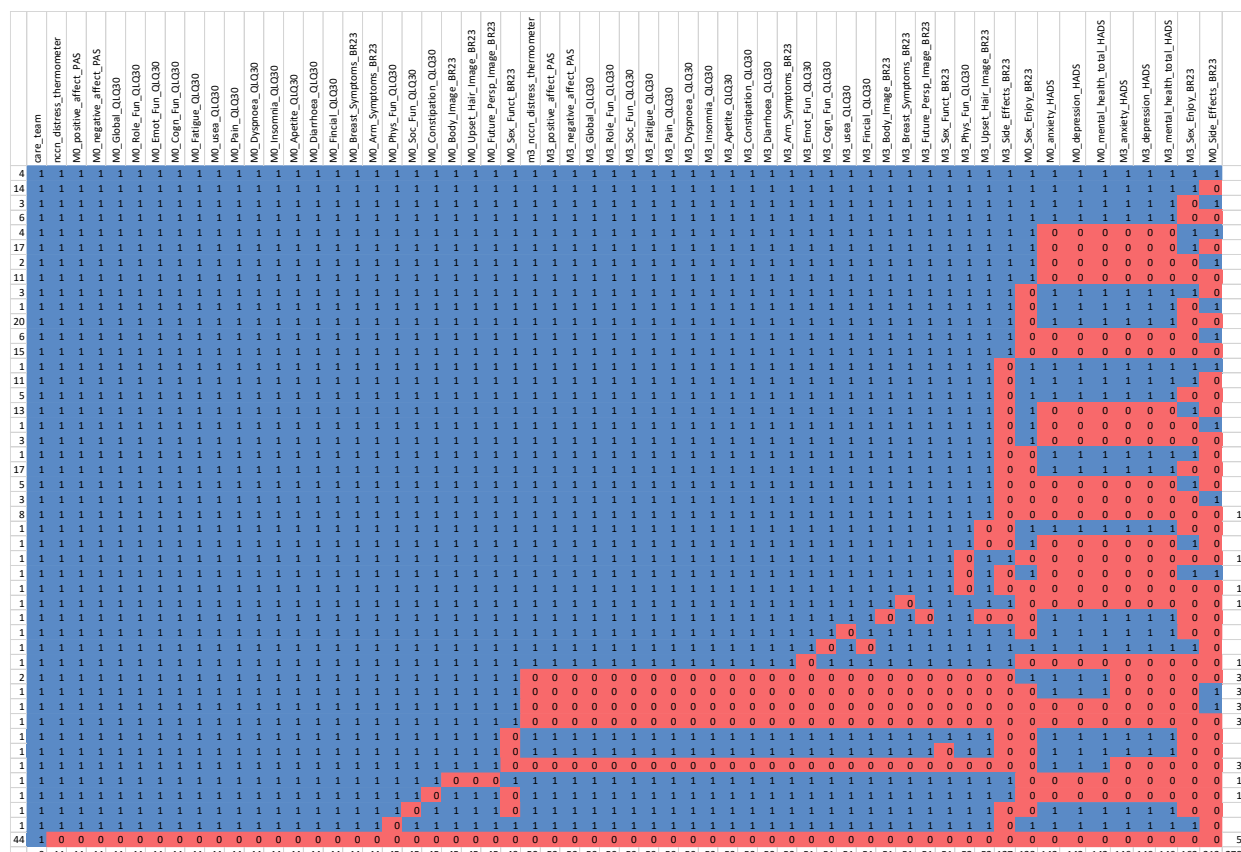


Figure 24: Example of analysis of missingness for HUS subset

Figure 24 depicts an analysis of missing values that was performed in a subset of a dataset from HUS. In detail, 44 patients with records in sociodemographic, lifestyle, treatment and clinical variables had consistently missing values in C30, BR23, HADS, and PANAS subscale scores. This strange pattern implied some type of interruption or corruption in the data flow. The diagram can be interpreted as follows:

- Column on the left (count): shows numbers of cases/patients in each pattern. Each row after count represent a missing pattern. Blue: observed, Red: Missing.
- Column on the right (count): shows numbers of items missing in each pattern.

During **diagnosis** phase, the true nature of the suspected data is clarified. Worrisome records and patterns are carefully examined and diagnosed either as erroneous, true or suspected (i.e., no definite explanation found, but still suspected). The source of anomalies and contradictions is tackled. The procedure followed is to go to previous stages of the data flow and examine whether the anomaly is present or not. To this end, indicative comparisons between Noona exports (CHAMP, IEO and HUS) or HUJI Qualtrics/Excel and the integrated dataset were performed. Part of the integrated dataset was recreated (outside the data lake) for variables with consistently suspected records to facilitate comparisons.

Sources of problems identified that are related to data include:

- **Inconsistent values** entered by the participant or by the clinical personnel (e.g. smoking=never, number of cigarettes=3),
- **Correct values filled out in wrong cells** by the participant or the clinical personnel entering the data (e.g. weight=167, height=75),
- **Differences in categorical variable codings** between Noona exports (CHAMP, IEO and HUS), and HUJI Qualtrics/Excels, mainly for sociodemographic and lifestyle variables,

(e.g. for “do_you_drink_now_less_the_same_amount_more_than_before_the_illness”: coding for HUJI dataset: 1=less 2=same amount 3=more; coding for Noona exports 1=less 2= more 3= same amount),

- **Differences in naming conventions and the structure of information** between Noona exports (CHAMP, IEO and HUS), and HUJI Qualtrics/Excels. This mainly applies for sociodemographic, lifestyle, treatment and clinical variables. E.g. for type of exercise (none/moderate aerobic/heavy aerobic/muscle training) one variable exists in Noona exports (that merges all participant’s answers) and four in HUJI datasets (one for each type of exercise),
- **Differences in units** between clinical sites (e.g. for blood tests, or one of the exercise time variables between Noona exports and HUJI datasets),
- For a single participant, the **order of data records** in Noona export files may not match the wave sequence, e.g. the records for month 3 may appear before baseline in the same file. This is the case when participants answer in paper-and-pencil and their answers are inserted in Noona by clinical personnel at a later time point.
- **Formatting problems**, e.g. free text in cells.
- **Duplicates** (exact or partial) when the same information is entered twice in Noona or a previously stored information is corrected.

After identification of errors, suspected records and strange missing patterns, problems are documented in detail and shared with the relevant partners (clinical and technical partners) for treatment or editing (**post-processing** phase). Whenever possible, faulty records are corrected by the clinical partners. Erroneous values if not corrected are deleted. The process has greatly contributed to the definition of the required rules and the optimization of the Data Cleaner and the integration/harmonization module on Data Lake.

5. BOUNCE Data Harmonization & Storage

Within the context of BOUNCE, the data harmonisation and storage process is composed of a set of clear and distinct steps towards the efficient and effective exposure of both relational and semantically annotated data which will be used as input in the data analysis. The detailed process for the harmonization and storage actions performed after cleaning is shown in Figure 25.



Figure 25: Data harmonization & storage process

The steps of the process are as follows:

- **Aggregate data in a single database:** As depicted in the figure, in the first step the cleaned data are transformed and loaded to a single database in order to ensure that all data between the four clinical centers that are dealing with a specific questionnaire are stored into the same table.
- **Recoding:** During this process, in many cases recoding is required as clinical centers might use different coding or measurement units. For example, neutrophiles for IEO and CHAMP are percentages and for the rest of the centers not percentages.
- **Relational API:** As soon as the data have been successfully recoded and aggregated into a single database, they are exposed through the relational API. Using the specific API the services of the BOUNCE platform that are aware of the specific structure of the database can issue queries directly to the relational database as long as they have the necessary permission to do so, which is ensured by the BOUNCE platform's security component, namely the Access Controller.
- **Semantic API:** Leveraging the available BOUNCE semantic model, in this step all relational data are mapped to the appropriate ontology terms. As such, the data are also made available through the Semantic API and can be used directly using the corresponding SPARQL queries. As with the Relational API, the API can be only invoked by the services with the appropriate permissions, as regulated by the Access Controller.

Currently the data shown in Figure 26 are available in the BOUNCE data repository including psychoemotional, sociodemographics, exercise data, tumour biology, therapy stages, genetic risk factors, etc.

Domain	Abbreviation	Measure name	M0	M3	M6	M9	M12	M15	M18
Personality	TIPi	Ten Item Personality Measure (brief "Big Five")	x						
	LOT-R	Optimism/Pessimism	x						
Meaning	SOC- 13	Sense of Coherence	x						
Trauma and PTSD	PCL-5	PTSD Check-List			x		x		x
		Recent negative life events	x	x	x	x	x	x	x
		Recent illness-related events		x	x	x	x	x	x
Coping	PACT	The Perceived Ability to Cope With Trauma (Flexibility in coping)	x			x		x	
	CERQ short	Cognitive Emotion Regulation Questionnaire	x			x		x	
		MAAS - Mindfulness	x				x		
		Spirituality coping - a visual bar		x		x		x	
Social Support	mMOS-SS	modified Medical Outcomes Study Social Support Survey		x		x		x	
	F.A. R.E.	1.Communication and cohesion; 2. Perceived family coping subscales		x		x		x	
		Instrumental/emotional perceived social support	x	x	x	x	x	x	x
Resilience	CD-R ISC	Connor-Davidson Resilience Scale	x			x		x	
		How much are you back to yourself?			x	x	x	x	x
Illness Perception	IPQ	Illness Perception Questionnaire			x		x		x
	B-IPQ	Items no 3 and 4 from B-IPQ		x	x	x	x	x	x
	mini-MAC	Mental Adjustment to Cancer		x		x		x	
		Single item: what has done to cope (open question)		x	x	x	x	x	x
	CBi-B	Cancer Behavior Inventory (self-efficacy in coping with cancer)	x		x		x		
		A general self-efficacy item	x	x	x	x	x	x	x
		Adherence to medical advice:item 5 from the MOSAdherence to medical		x	x	x	x		x
	PTGI	The Posttraumatic Growth Inventory - short form		x			x		x
Quality of life	QLQ-C30	EORTC quality of life questionnaire	x	x	x	x	x	x	x
	QLQ-BR23	EORTC quality of life questionnaire breast cancer module	x	x	x	x	x	x	x
Distress	FCR I-SF	Fear of Recurrence - short form (severity scale of original FCRl)	x		x		x		x
	HADS	Hospital Anxiety and Depression Scale	x	x	x	x	x	x	x
	DT	NCCN Distress Thermometer	x	x	x	x	x	x	x
	PANAS	Positive and Negative affectivity - short form	x	x	x	x	x	x	x
		Age	x						x
		Highest level of education	x						x
		Marital status	x						x
		Number of children	x						x
		Employment status	x	x	x	x	x	x	x
		Income	x						x
		Absence from work	x	x	x	x	x	x	x
		Smoking and alcohol/drugs consumption	x		x	x	x		x
		Weight and height	x		x		x		x
		Diet	x			x			x
		Exercise	x	x	x	x	x	x	x
		Number of counselling/support sessions	x	x	x	x	x	x	x
		Number of visits with physician/nurse/social worker	x	x	x	x	x	x	x
		ICD-10 Classification	x						
		Tumor biology	x				x		
		Surgery type and side	x				x		
		Performance status	x	x	x		x		
		Previous/ongoing oncological therapy	x				x		x
		Menopausal status (pre-menopausal, peri-menopausal, post menopausal)	x				x		
		Genetic risk factors	x				x		
		Use of psychotropic medication	x	x	x		x		
		Basic laboratory tests (blood cell counts, hs-CRP) at baseline and at Month 12	x				x		
		Other chronic illnesses	x				x		

Figure 26: An overview of the BOUNCE dataset available in the BOUNCE data repository.

Similarly for the external trial dataset provided by IEO the following data are included as shown in Figure 27, including psychological measures, sociodemographics and medical and treatment data.

	Domain	Abbreviation	Measure name	M0	M3	M6	M12	M24
Psychological Measures	Personality	LOT-R	Optimism/Pessimism	x	x	x	x	x
	Resilience	RSA	Resilience Scale for Adults	x	x	x	x	x
	Illness Perception	B-IPQ	Items no 3 and 4 from B-IPQ	x	x	x	x	x
		CBI-B	Cancer Behavior Inventory (self-efficacy in	x	x	x	x	x
	Quality of life	QLQ-C30	EORTC quality of life questionnaire	x	x	x	x	x
		QLQ-BR23	EORTC quality of life questionnaire breast	x	x	x	x	x
Sociodemographics	Age			x				
	Weight and height			x				
medical and treatment data	ICD-10 Classification			x				
	Tumor biology			x				
	Surgery type and side			x				
	Previous/ongoing oncological therapy			x				
	Menopausal status (pre-menopausal, peri-menopausal, post menopausal)			x				
	Genetic risk factors			x				
	Basic laboratory tests			x				

Figure 27: An overview of the IEO external dataset available in the BOUNCE data repository.

5.1. Semantic Data Mapping Overview

In this subsection, the process and the tools used for exposing the data collected through the Semantic API are described in detail. There are many techniques and tools to conduct semantic data mapping and access relational databases as RDF Graphs. In this work, we used the D2RQ Platform¹ which consists of the D2RQ Engine, the D2R server and the D2RQ mapping language as its main components.

Figure 28 displays the architecture of the D2R platform. There are three main parts of D2RQ that can collaborate and communicate with external applications and systems. The first part is the D2RQ engine, which handles the communication with any relational database. It contains the RDF Dump component which is responsible for exporting custom dumps of the database in RDF format, and for loading them in RDF Stores. The second part is the D2RQ mapping file, which is used for the data integration part. The D2RQ file (in .ttl file format) is used to describe the relation between an ontology and a relational data model by specifying the declarative mappings from the database tables and columns to the RDF classes and properties. Finally, the third part is the D2R server, which is an HTTP server that provides a Linked Data view, an HTML view for debugging and a SPARQL Protocol endpoint over the database.

¹ D2RG, <http://d2rq.org/>

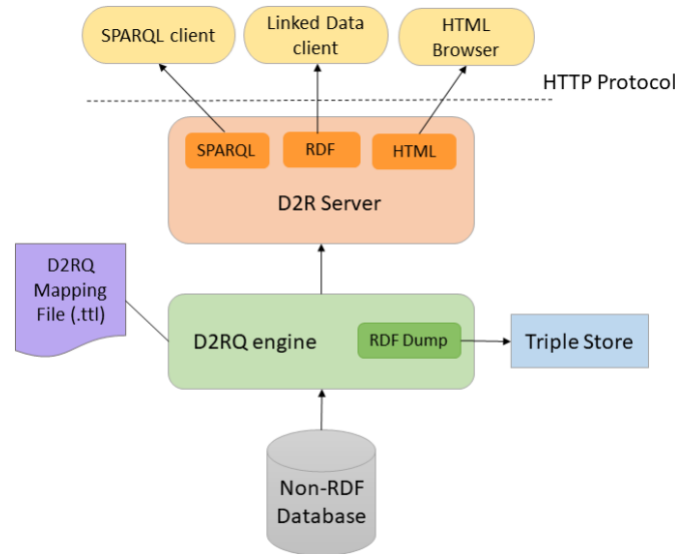


Figure 28: D2RQ Architecture

D2RQ for the BOUNCE Questionnaire Data. The first step for exposing the available data within the BOUNCE platform as linked data is to generate the necessary D2RQ mapping file. The D2RQ mapping file is itself an RDF document written in Turtle syntax. The mapping is expressed using terms in the D2RQ namespace: <http://www.wiwiss.fuberlin.de/suhl/bizer/D2RQ/0.1#>

The following example maps part of the database schema to RDF, based on the BOUNCE ontology.

```

map:TIPI a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:class bpo:TIPI_Ten_Item_Personality_Measure;
  d2rq:uriSqlExpression "CONCAT('http://localhost:2020/resource/TIPI/',
    integrated.FORTH_id)".

map:Openness a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:class bpo:openness;
  d2rq:uriSqlExpression "CONCAT('http://localhost:2020/resource/TIPI/Openness/',
    integrated.FORTH_id)".

map:TIPI_measures_openness a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:TIPI;
  d2rq:property bpo:measures_big_five_personality_domains;
  d2rq:refersToClassMap map:Openness.

map:mo_openness_TIPI a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:Openness;
  d2rq:property bpo:has_value;
  d2rq:column "integrated.M0_openness_TIPI";
  d2rq:datatype xsd:float.
  
```

A ClassMap is used to map each row of the table, representing each patient by their FORTH_id, to one RDF resource identified by a URI like '<http://localhost:2020/resource/TIPI/42>', where 42 is the row's primary key value (i.e. patient's unique identifier) and TIPI is the questionnaire answered by that specific patient. Each of these resources is of rdf:type bpo:Questionnaire (e.g. for the TIPI questionnaire the rdf:type is bpo:TIPI_Ten_Item_Personality_Measure). Another

ClassMap is getting created afterwards, representing the scale measured by the specific questionnaire (e.g. bpo: Openness).

Then, a PropertyBridge is used to attach a bpo:measures object property to each resource according to the questionnaire it refers to (e.g. bpo:measures_big_five_personality_domains for the TIPI questionnaire), whose domain is the questionnaire ClassMap (e.g. TIPI) and range the scale measurement ClassMap (e.g. Openness). A second PropertyBridge attaches the information retrieved from the corresponding column (e.g. MO_openess_TIPI) of the SQL table. Figure 29 illustrates the structure of a D2RQ map example, for the TIPI questionnaire, and the Openness and Extraversion scale measurements (the other 3 scales were omitted for readability purposes).

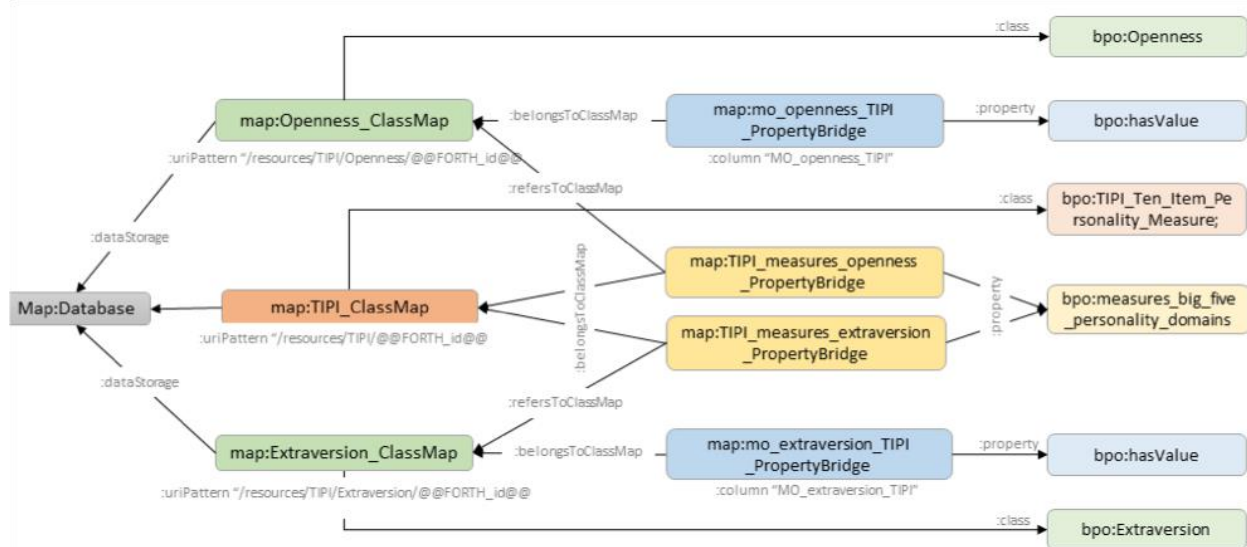


Figure 29: Structure of an example D2RQ map

After using the mapping file and the D2R Server tools, the relational database gets published in RDF format, with SPARQL support. Figure 30 illustrates an RDF graph example for the TIPI questionnaire. As all terms available in the BOUNCE dataset were also present in the external trial dataset provided by IEO, it was similarly mapped to the ontology already available and subsequently exposed through the semantic API.

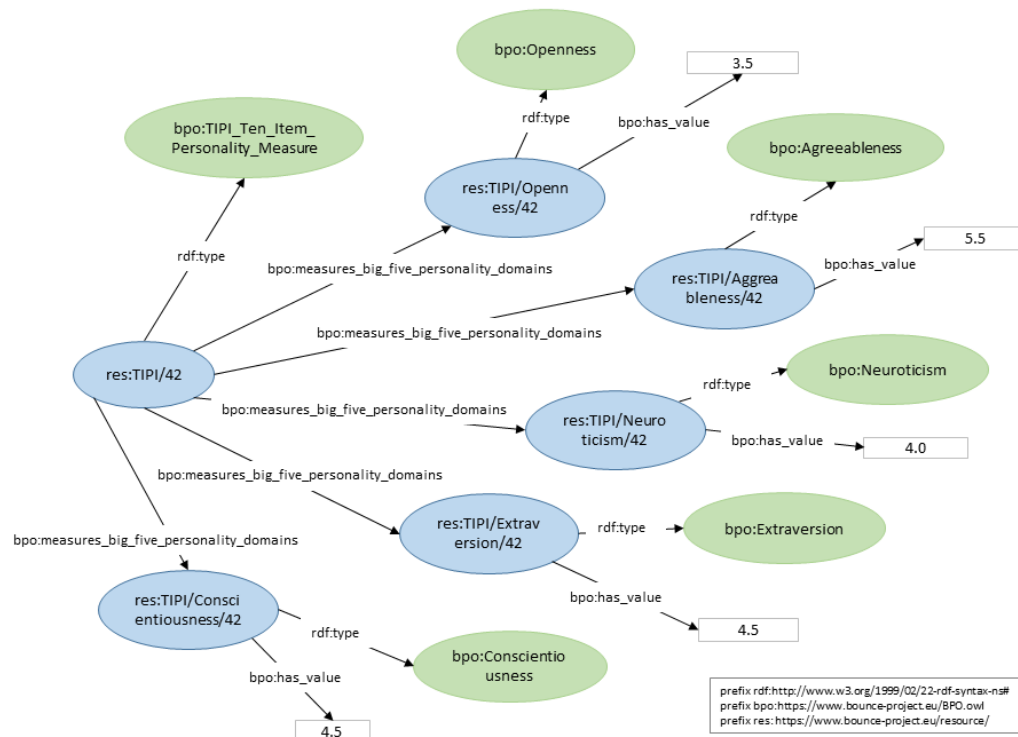


Figure 30: RDF Graph example about the TIPI questionnaire

6. Conclusions

The scope of deliverable D3.4 “Solutions for Data Aggregation, Cleaning, Harmonization & Storage” was to document the efforts undertaken within the context of Task 3.3 - Data Source Aggregation & Cleaning and Task 3.4 - Data Source Harmonization & Storage. The deliverable reported the detailed documentation with regards to: a) the BOUNCE data aggregation process that is followed during the collection and aggregation phase of the prospective data in the clinical centres, b) the BOUNCE Data Cleaning process that is performed on top of all the collected and aggregated data (both from the clinical centres and the external trial data provided by IEO) in two stages, the automatic cleaning process and the enhanced data cleaning process and c) the BOUNCE Data Harmonization and Storage process which is performed on all the “cleaned” data towards their exposure as both relational and semantically annotated data that will be used as input the data analysis of the BOUNCE platform.

More specifically, the current deliverable documented BOUNCE data aggregation process which incorporates the steps from the data collection of core and complementary study-specific data as provided by the patients and the clinical staff, to their effective and efficient aggregation and export and their initial insertion into the BOUNCE Data Lake. The deliverable presented the two complementary applications that are provided by the Noona ecosystem which facilitate this process. At first, the Noona Core application, consisting of two parts, the patient application and the clinic application was documented. The deliverable presented the patient application that is offered as both a web and a mobile application and is leveraged by the patients in order to communicate with their clinics. Furthermore, the Noona clinic application that is utilised by the clinical staff to report the daily treatment tasks was also presented. Finally, the deliverable presented the Noona eCRF application, which constitutes the offered application for the complementary study-specific data collection. Following the applications documentation, the deliverable reported the comprehensive documentation of the applied, during the collection and aggregation process, data quality controls where for each input type the relevant data quality checks are documented, as well as the offered data export process which is leveraged for the uploading of the data to the BOUNCE Data Lake.

Following the BOUNCE data aggregation process, the deliverable presented the holistic BOUNCE Data Cleaning process which is composed of the automatic cleaning process and the enhanced data cleaning process. In detail, the deliverable presented the details of the designed automatic data cleaning process which includes the preliminary data analysis, the definition of data validation, data cleaning and data completion aspects of the data cleaning workflow, as well as its execution and verification. The deliverable presented also the implementation details of the BOUNCE Data Cleaner that realises this process by adopting the concept of data validation, data cleaning and data completion rules. Moreover, a set of examples that demonstrate the usage of the BOUNCE Data Cleaner were presented for both the prospective data in the clinical centres and the data from the external trial provided by IEO. The deliverable presented also the details of the enhanced data cleaning process and its three distinct phases, namely the screening, diagnosing and post-processing phases. For each phase, the performed steps and corrective actions that were performed during their execution were presented in detail. In addition to this, a set of examples illustrating the utilisation of the process during the holistic BOUNCE Data Cleaning process was presented.

Finally, the deliverable documented the details of the BOUNCE Data Harmonization and storage process. During this process, the data originating from both the clinical centres and from the external trial as provided by IEO are aggregated into a single database and they are recoded in order to be eventually exposed through a relational API and a semantic API. The deliverable

presented in detail the semantic data mapping process in which the provided are semantically annotated based on the BOUNCE ontology before they are published in RDF format, with SPARQL support. In addition to this, the deliverable presented an overview of the datasets which are available on BOUNCE platform.

The deliverable concludes the activities of WP3 per the BOUNCE description of Action and constitutes the final report of the outcomes of the specific work package. It provides the final documentation of the processes and workflows that are leveraged in BOUNCE in order to produce the required interoperable version of the collected data.

7. References

- [1] “Deliverable:1.3 BOUNCE methodology,” The BOUNCE Consortium, July 2018.
- [2] “Deliverable:6.1 Clinical pilot methodology and preparatory actions,” The BOUNCE Consortium, October 2018.
- [3] J. C. S. E. R. a. H. K. Van den Broeck, “Data cleaning: detecting, diagnosing, and editing data abnormalities,” *PLoS Med*, vol. 2, no. 10, 2005.
- [4] W. Winkler, “Problems with inliers,” Research Reports Series RR98/05, 1998. [Online]. Available: <https://www.census.gov/content/dam/Census/library/working-papers/1998/adrm/rr9805.pdf>. [Accessed 20 July 2021].