uc3m | Universidad Carlos III de Madrid

Bachelor's Degree in Biomedical Engineering

2018-2019

Bachelor Thesis

"Integrated IT tools for the management and exploitation of European paediatric transplantation data"

Gonzalo Sofío Toro

Javier Pascau González Garzón

Juan Manuel Torres Canizales

Leganés, March 2019

ABSTRACT

This thesis was developed with the aim of building and implementing new informatic tools that could facilitate research and international benchmarking in paediatric transplantation.

Two different studies were created for that purpose. The first one consisted on an analysis of the current paediatric transplantation activity situation in the European Union, utilizing Google My Maps to obtain a general geographical distribution per countries, and Data Studio to complement the previous one and get more complete results. The second method was the construction of a patient registry for TransplantChild, the European Reference Network for paediatric transplantation, so that all the transplanted children in the member hospitals are registered, and their information is stored together to be analysed.

The results obtained give an accurate vision of the unequal transplantation distribution across the continent and allowed the identification of the most expert and specialized centres in Europe. Also, it was possible to recognize potential cases in which the child needed to be moved to receive a transplant, and proposed solutions for them. On the other hand, two platforms, one for data collection and one for data exploitation, were integrated together to build up the patient registry. The latter was developed from scratch in this project using Python and Flask.

At last, it was concluded that by implementing the mentioned tools it is possible to promote paediatric transplantation research and perform a benchmarking across the countries and hospitals. Hence, this turns out to be an indirect way to improve transplants success rate and, in the end, patients' survival and quality of life.

Keywords: Paediatric transplantation activity; European Reference Network; Google My Maps; Data Studio; Data Analysis; Patient registry; Python; Flask.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my parents, my little brothers, and my girlfriend for all their support they gave me during the development of this thesis. Thank you for standing me in my worst moments, when I felt overwhelmed by the project and the situation. You gave me the necessary strength and calm to keep working no matter what.

Also, I would like to thank both of my tutors, who helped me with everything they could and were always there every time I needed them. Thanks for supporting and advising me.

I appreciate all the help that was offered to me regarding the project development. Thank you very much to everyone that contributed for it to be a little bit better.

INDEX

FIGURE INDEX

TABLE INDEX

LIST OF ABBREVIATIONS

API	Application Programming Interface
BSD	Berkeley Software Distribution
DAG	Data Access Group
ERN	European Reference Network
EBMT	European society for Blood and Marrow Transplantation
EU	European Union
EUCERD	European Union Committee of Experts on Rare Diseases
GDPR	General Data Protection Regulation
GvHD	Graft versus Host Diseases
HCP	Health Care Provider
ISCIII	Health Institute Carlos III
HSCT	Hematopoietic Stem Cell Transplant
IdiPAZ	Hospital La Paz Institute for Health Research
JSON	JavaScript Object Notation
ONT	Organización Nacional de Trasplantes
PT	Paediatric Transplantation
PETER	PaEdiatric Transplantation European Registry
PTLD	Post-Transplant Lymphoproliferative Disease
PTSD	Post-Traumatic Stress Disorder
PPRL	Privacy Preserving Record Linkage
PyPI	Python Package Index
PSF	Python Software Foundation
REDCap	Research Electronic Data Capture
SOT	Solid Organ Transplant
TTP	Trusted Third Party
URL	Uniform Resource Locator
UK	United Kingdom
WHO	World Health Organization

1. INTRODUCTION

1.1. Motivation

Medical investigation has always been oriented to increase patients' survival and improve their quality of life as much as possible. Despite this research, throughout history, millions of people have died as a result of their medical conditions that, at that determined time, were considered incurable. However, with the development of healthcare, it was managed to treat many of these diseases.

An example for this is smallpox, which was a widely spread illness that caused around 400,000 deaths per year in Europe in the last years of the 18th century [1]. Thanks to the investment on its research, this condition was declared eradicated by the World Health Organization (WHO) in 1979 [2].

Nevertheless, transplants research has a different characteristic to any specific disease investigation. This is that, if the transplantation method is perfected and its success is guaranteed, it would impact the treatment of many conditions that have this technique as their only remedy nowadays. In order to help this research, it is necessary to analyse the current transplants data with the objective of identifying new investigation areas related to such a complex procedure.

Moreover, even though the procedures and treatments are similar for adult and Paediatric Transplantation (PT), there exist some important differences that make the latter even more complicated than the former. Children have to live together with the graph for longer, so they can suffer more post-transplant complications, and growth is also a factor that needs to be considered when dealing with paediatric patients because, for example, immunosuppression, can interfere with it [3].

As the cases of PT are very low when compared to adult transplants performed, usually the latter are investigated instead of the former. However, researching in PT is a high-rewarded task: most of the results obtained can be applied to adults, but not vice versa since transplantation in children has characteristic aspects that need to be investigated independently.

Because of the previous circumstances, ERN TransplantChild, hosted in La Paz University Hospital, which is one of the reference centres in Spain and Europe, decided to take a step forward and develop a project consisting on different tools to analyse the current PT situation and implement an own registry for the network. The goal behind this project was to promote data collection, registration and benchmarking between countries and transplantation centres.

Hence, this thesis finds its motivation in helping to facilitate research in PT as previously described, with the aim of improving survival and quality of life, especially in the long term, for children who need a transplant.

1.2. Objectives

The project consisted in implementing two different ways of PT data analysis and exploitation. Even though these methods were carried out independently, they share a common objective and will be integrated together in the future.

Specific objectives were established with the purpose of optimizing the study’s helpfulness regarding transplantation research:

- Elaborate a complete analysis of the PT activity in the European Union (EU), containing the following aspects: hospitals geographical distribution, identification of expert centres and relevance of TransplantChild inside the continent.
- Identify possible solutions to improve the PT situation in the EU.
- Inside the PaEdiatric Transplantation European Registry (PETER) developed by TransplantChild, finish the creation of a data collection platform, and develop a new one for data exploitation.

1.3. Time plan

In order to obtain a better thesis organization and therefore better results, a time plan was elaborated so that all the different work packages were scheduled and programmed. Each one was divided in more specific tasks to obtain a more complete time plan in which all the parts of the project could be identified.

Apart from that, in the planning, the study milestones were set. They represent the date in which every assignment had to be finished. Then, with this information, the time plan was obtained by creating a Gantt chart comprehending from September 2018 to February 2019, as shown in Table 1.1.

TABLE 1.1. PROJECT TIME PLAN GANTT CHART.

Month number	-1	1	2	3	4	5	+1
Month name	Sep	Oct	Nov	Dec	Jan	Feb	Mar
Year	2018	2018	2018	2018	2019	2019	2019
A.1. Analysis of paediatric transplantation activity in Europe	Start						End
T.1.1. Data organization	Start		End				
T.1.2. Map construcion	Start			End			
T.1.3. Data reports		Start				End	
T.1.4. Data analysis				Start			End
M.1. Map implementation			M.1				
M.2. Reports implementation					M.2		
A.2. Development of a platform for data exploitation	Start						End
T.2.1. Features identification	Start		End				
T.2.2. Data collection platform evaluation	Start			End			
T.2.3. Data exploitation platform exploring	Start			End			
T.2.4. Platform development			Start				End
M.3. Platform implementation						M.3	
A.3. Bachelor thesis report elaboration			Start				End
T.3.1. References finding & structure			Start		End		
T.3.2. Introduction writing			Start			End	
T.3.3. General writing				Start			End
T.3.2. Revising					Start		End
M.4 Document submission						M.4	

Activity	Start
Task	End
	Milestone

Source: Project planification.

1.4. Structure

1.4.1. Project structure

In the block diagram in Fig 1.1 are represented all the tasks assigned to the project. It includes extra work packages that were not included in it, but they enhance the easy understanding of the scheme. The shapes correspond to different elements, and the element types are identified with colours.

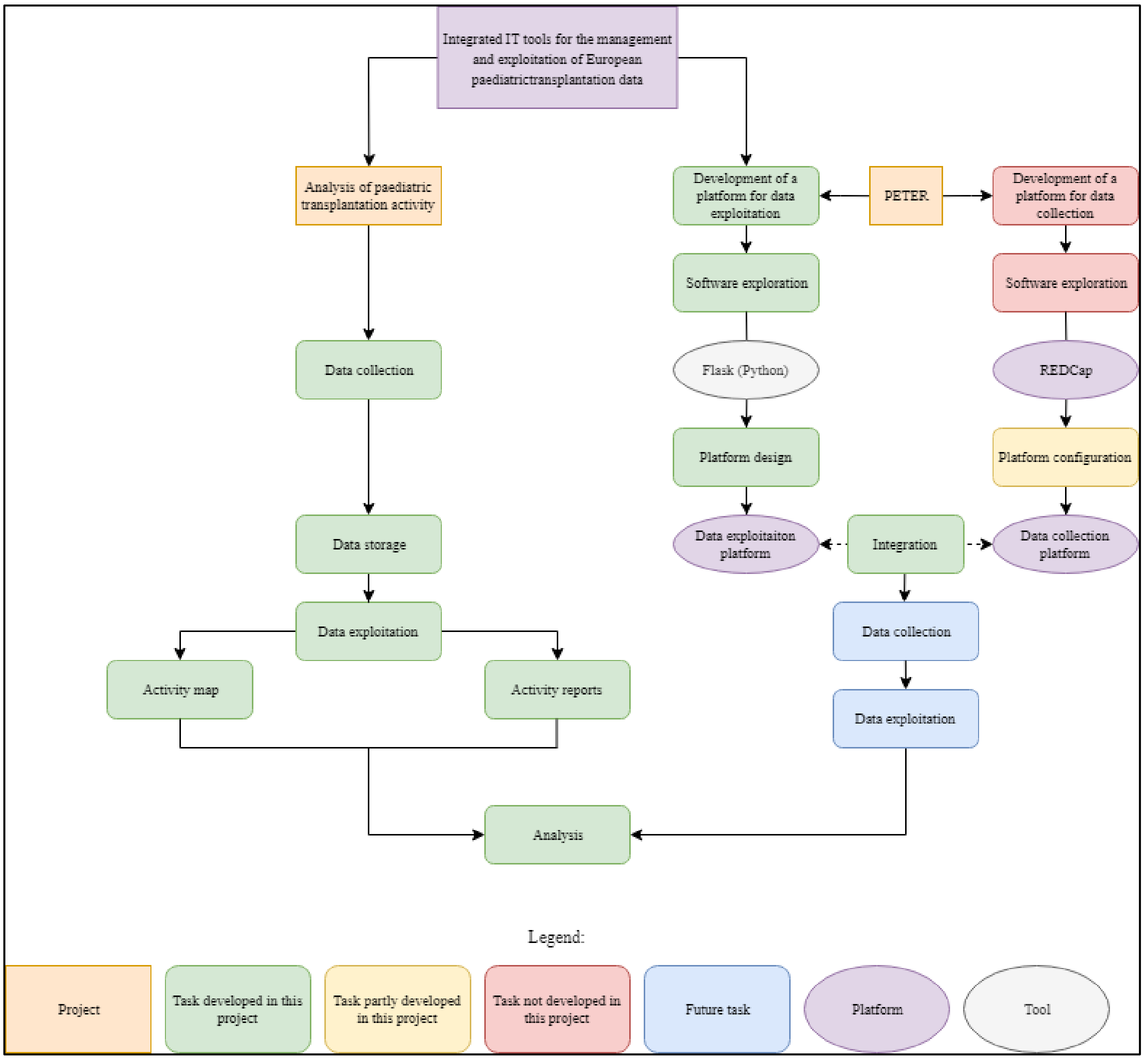


Fig 1.1. Project block diagram.

1.4.2. Document structure

This document has been structured in four different chapters that describe in detail the process followed to achieve the mentioned objectives and give the necessary information for the proper project understanding.

The Introduction presents the reason behind the realization of this thesis, in the sections of Motivation and Objectives. Apart from this, it serves as a background to introduce the most technical concepts utilized, so that the rest of the document is easy to understand.

In the second chapter, Materials & Methods, all the needed tools and software programs are presented and explained with their respective features. Moreover, the methods and techniques utilized are detailed. Later, in Results, the obtained outputs from the previous chapter are shown and interpreted. Everything is divided in two different sections corresponding to the works carried out: Analysis of PT activity in the European Union and PETER platform development.

Then, in Discussion, all the legal and social aspects of the undertaking are stated, followed by their limitations and ways to improve it in the future. To finish, there is a last section for analysing the conclusions achieved.

1.5. Background

1.5.1. Transplantation

In medicine, transplantation is defined as the surgical replacement of a damaged or non-functioning organ or tissue by a healthy one. The implanted graft can proceed from the same patient or from a donor, which can be living or cadaveric.

Nevertheless, a transplant is not just a surgery, as shown in Fig 1.2. For instance, when a patient is diagnosed to receive a transplant, it is included on the waiting list until a compatible donor is found [4]. Then, after the operation, it is necessary to maintain an immunosuppression treatment, in order to minimize the chances of rejecting the graft. Despite this, many complications may appear in the long term, such as Graft versus Host

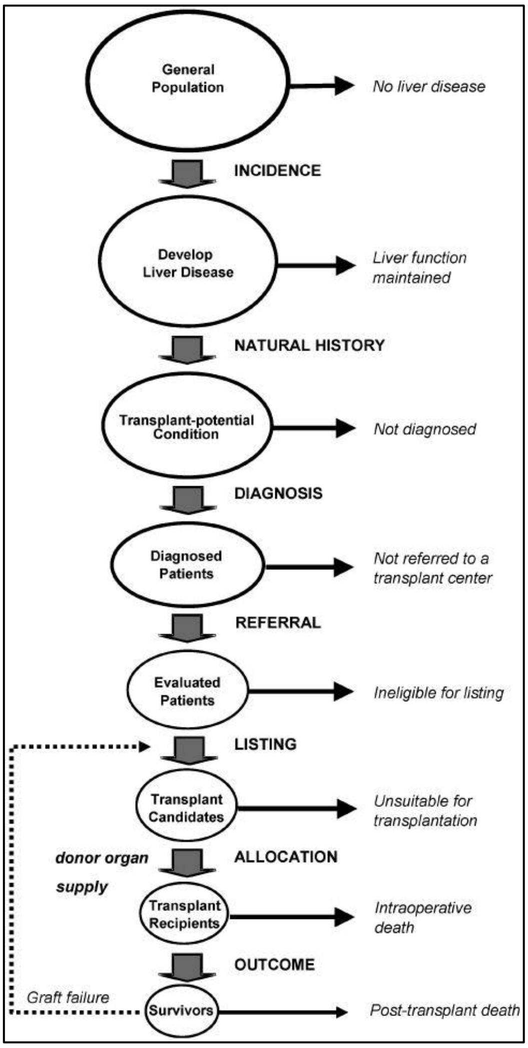


Fig 1.2. Liver transplantation process scheme [4].

Disease (GvHD), infection or Post-Transplant Lymphoproliferative Disease (PTLD) [5].

1.5.1.1. Transplant classification

Depending on the transplanted organ or tissue, transplants are divided in two types:

– Hematopoietic Stem Cells Transplants (HSCTs): Stem cells are defined as an undifferentiated type of cells which are able to renew themselves indefinitely and generate highly specialized cells. Specifically, hematopoietic stem cells can differentiate into any mature blood lineage [6]. Due to these properties, HSCTs are performed to patients with any defect in the immune system, or for metabolic or blood disorders [7]. An HSCT can be autologous, if the cells to be transplanted belong to the own patient that receives them, or allogeneic if they come from a different individual [6]. Autologous HSCTs are usually preferred for autoimmune diseases [8], whereas allogeneic transplants are most commonly used to treat haematological malignancies [6]. The main indications for each type of HSCT are represented in Fig 1.3.

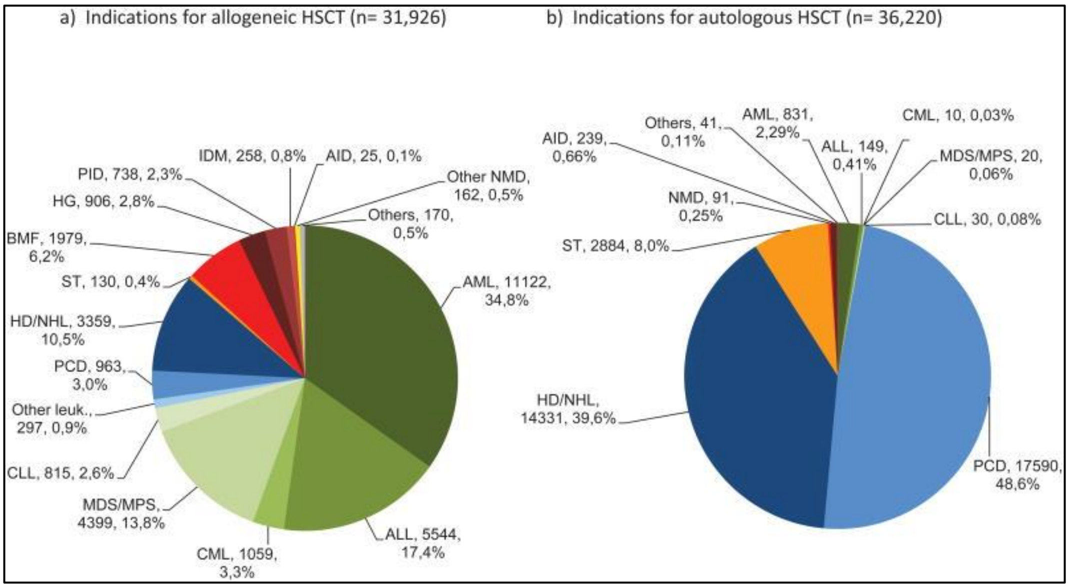


Fig 1.3. Proportions of indications for HSCT [65].

AML: Acute Myeloid Leukaemia. ALL: Acute Lymphoblastic Leukaemia. CML: Chronic Myeloid Leukaemia. MDS/MPS: Myelodysplastic/ Myeloproliferative Syndromes. CLL: Chronic Lymphoproliferative Leukaemia. PCD: Plasma Cell Disorders. HD/NHL: Hodgkin Disease/ Non-Hodgkin Lymphoma. ST: Solid tumours. BMF: Bone Marrow Failure. HG: Hemoglobinopathies. PID: Primary Immune Deficiencies. IDM: Inherited Disorders of Metabolism. AID: Auto Immune Disease. NMD: Non-Malignant Disorders.

- Solid Organ Transplants (SOTs): In this case, an organ (or more than one) is transplanted because the patients' one has partially or totally lost its function. Possible SOTs are liver, kidney, heart, lung, intestinal and multivisceral. However, these last two are usually overlapping, because intestinal transplants are frequently performed together with more organs, such as the liver, stomach or pancreas [9]. Hence, for this undertaking, intestinal transplants were included inside the category of multivisceral.

1.5.1.2. Paediatric transplantation

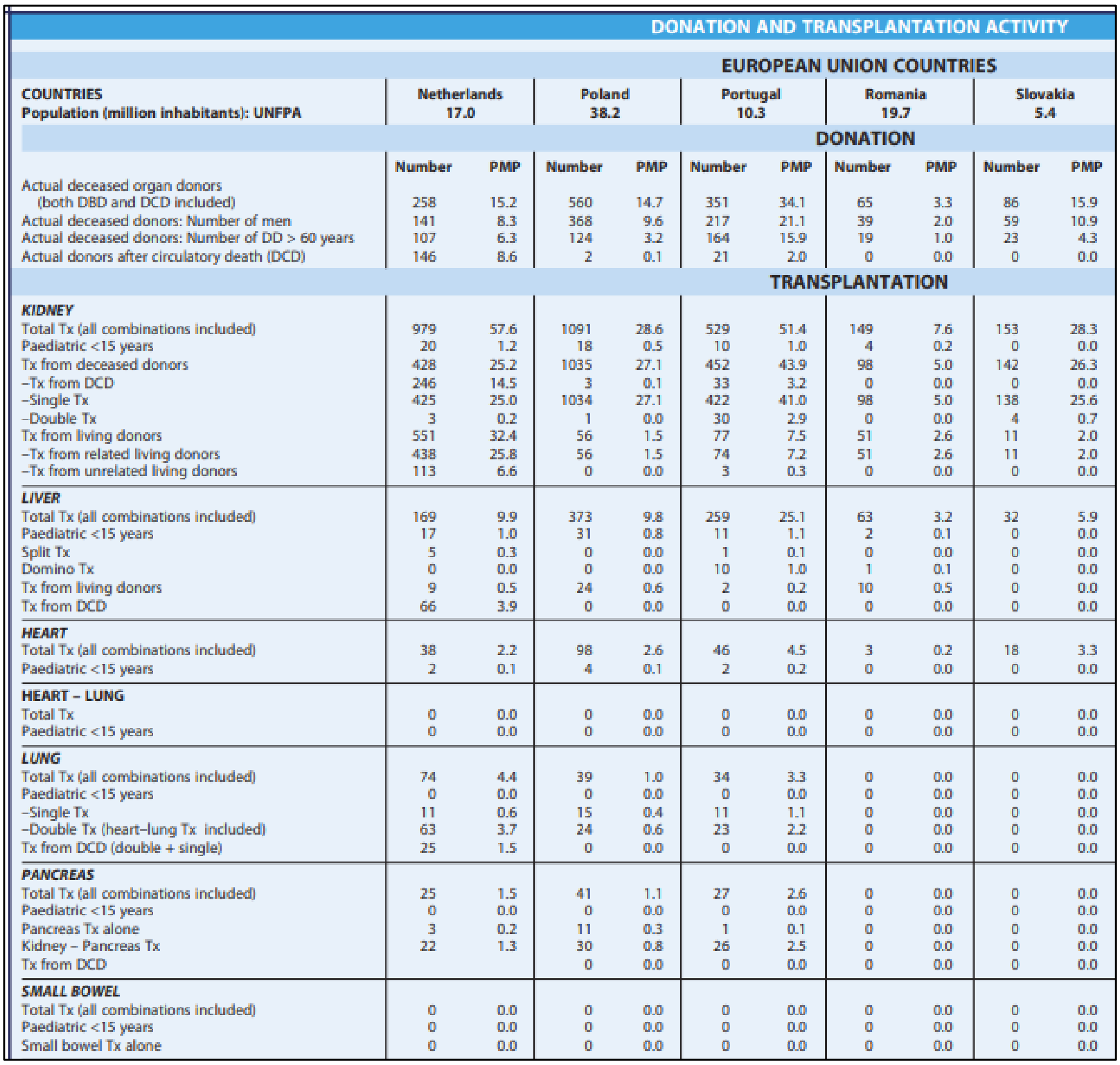
When a transplant is performed on a child, it is defined as PT. This concept can vary depending on the country, since they can independently set the age limit for a patient to be considered paediatric. However, there is a general agreement to consider the paediatric age limit as 18. In fact, approximately half of the countries belonging to the EU have established this frontier. In the rest of countries, there are different values, always varying between 14 and 19 [10]. Therefore, in this project, the paediatric age limit was set to 18 years old.

When treating children, it is crucial to consider all the specific aspects about PT that make it different from adult transplantation. There are several factors, like physiology, donor factors and graft availability, pharmacokinetics and immunosuppression, among others [11]. Immunosuppression is one of the most evident differences. If data about immunosuppressive drugs applied to transplanted patients are compared, there are significant differences. For example, in 2002 in the US, only 2% of the paediatric heart recipients were discharged using rapamycin, whereas for adults it was used in 10% of the patients [11].

1.5.1.3. Transplantation activity

Because of the complexity of transplantation (and especially PT), it is necessary to keep track of all the transplants performed. Globally, around 120,000 SOTs transplants and 50,000 HSCTs are performed every year. From them, paediatric transplants have a prevalence close to the 10% in SOTs and 20% in HSCTs [12].

The Council of Europe collects the European and global information in a newsletter published yearly. The data is registered in tables divided by country and transplant. However, PT activity collection is limited. In Fig 1.4, the presence of specific lines for paediatric patients[1] can be appreciated. Nevertheless, following this table in the newsletter there are data about waiting lists and family refusals and, in those cases, paediatric information is not present. It makes difficult researching on PT, as waiting lists are important when getting to conclusions about the current situation.



DONATION AND TRANSPLANTATION ACTIVITY

EUROPEAN UNION COUNTRIES

COUNTRIES **Population (million inhabitants): UNFPA**	**Netherlands** **17.0**		**Poland** **38.2**		**Portugal** **10.3**		**Romania** **19.7**		**Slovakia** **5.4**	
DONATION										
	Number	**PMP**	**Number**	**PMP**	**Number**	**PMP**	**Number**	**PMP**	**Number**	**PMP**
Actual deceased organ donors (both DBD and DCD included)	258	15.2	560	14.7	351	34.1	65	3.3	86	15.9
Actual deceased donors: Number of men	141	8.3	368	9.6	217	21.1	39	2.0	59	10.9
Actual deceased donors: Number of DD > 60 years	107	6.3	124	3.2	164	15.9	19	1.0	23	4.3
Actual donors after circulatory death (DCD)	146	8.6	2	0.1	21	2.0	0	0.0	0	0.0
TRANSPLANTATION										
KIDNEY										
Total Tx (all combinations included)	979	57.6	1091	28.6	529	51.4	149	7.6	153	28.3
Paediatric <15 years	20	1.2	18	0.5	10	1.0	4	0.2	0	0.0
Tx from deceased donors	428	25.2	1035	27.1	452	43.9	98	5.0	142	26.3
–Tx from DCD	246	14.5	3	0.1	33	3.2	0	0.0	0	0.0
–Single Tx	425	25.0	1034	27.1	422	41.0	98	5.0	138	25.6
–Double Tx	3	0.2	1	0.0	30	2.9	0	0.0	4	0.7
Tx from living donors	551	32.4	56	1.5	77	7.5	51	2.6	11	2.0
–Tx from related living donors	438	25.8	56	1.5	74	7.2	51	2.6	11	2.0
–Tx from unrelated living donors	113	6.6	0	0.0	3	0.3	0	0.0	0	0.0
LIVER										
Total Tx (all combinations included)	169	9.9	373	9.8	259	25.1	63	3.2	32	5.9
Paediatric <15 years	17	1.0	31	0.8	11	1.1	2	0.1	0	0.0
Split Tx	5	0.3	0	0.0	1	0.1	0	0.0	0	0.0
Domino Tx	0	0.0	0	0.0	10	1.0	1	0.1	0	0.0
Tx from living donors	9	0.5	24	0.6	2	0.2	10	0.5	0	0.0
Tx from DCD	66	3.9	0	0.0	0	0.0	0	0.0	0	0.0
HEART										
Total Tx (all combinations included)	38	2.2	98	2.6	46	4.5	3	0.2	18	3.3
Paediatric <15 years	2	0.1	4	0.1	2	0.2	0	0.0	0	0.0
HEART – LUNG										
Total Tx	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0
Paediatric <15 years	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0
LUNG										
Total Tx (all combinations included)	74	4.4	39	1.0	34	3.3	0	0.0	0	0.0
Paediatric <15 years	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0
–Single Tx	11	0.6	15	0.4	11	1.1	0	0.0	0	0.0
–Double Tx (heart–lung Tx included)	63	3.7	24	0.6	23	2.2	0	0.0	0	0.0
Tx from DCD (double + single)	25	1.5	0	0.0	0	0.0	0	0.0	0	0.0
PANCREAS										
Total Tx (all combinations included)	25	1.5	41	1.1	27	2.6	0	0.0	0	0.0
Paediatric <15 years	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0
Pancreas Tx alone	3	0.2	11	0.3	1	0.1	0	0.0	0	0.0
Kidney – Pancreas Tx	22	1.3	30	0.8	26	2.5	0	0.0	0	0.0
Tx from DCD			0	0.0	0	0.0	0	0.0	0	0.0
SMALL BOWEL										
Total Tx (all combinations included)	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0
Paediatric <15 years	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0
Small bowel Tx alone	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0

Fig 1.4. Example of data per country obtained from the transplant newsletter [21].

---

[1] In the newsletter, patients are considered paediatric if they are younger than 15 years old.

1.5.2. Registries

A registry is defined as "an organized system that collects, analyses, and disseminates the data and information on a group of people defined by a particular disease, condition, exposure, or health-related service, and that serves any predetermined scientific, clinical or/and public health (policy) purposes" [13]. Moreover, clinical registries can belong to various subtypes: disease, product and health service registries.

A disease (or condition) registry is characterized by being focused on patients with the same diagnosis. In the same way, a product one is based on the utilization of a common device or medicine, and if the shared aspect is a surgery, procedure or hospitalization then it is classified as a health service registry [13].

In this thesis, the centre of attention were the transplantation registries, and more precisely the paediatric ones. However, they do not belong to any of the abovementioned categories, as they share aspects with both disease and health service registries. The transplantation process contains the surgery, which is included in the health service registries, and the follow-up, which belongs to the disease type registries because of the chronic illness condition of transplants.

Registries can also be categorized according to their objectives. For instance, a registry can be focused on clinical data, quality of life, socioeconomic data, etc. In the case of this undertaking, the designed integrates more than one type: clinical, life expectancy, life quality and personal, according to the collected data, that contains variables from all of them.

One of the key functions of a registry is to identify challenges, necessities and new ways to improve the treatment of a determined disease (transplantation, in this case). To fulfil this function, it is crucial to contrast data from different sources. Therefore, in general, results are more productive when a registry is elaborated utilizing information from several hospitals situated in different countries, than when it is constructed with patients from a single health centre or country. Because of this, hospital networks are formed to facilitate research and share knowledge about difficult patients or conditions. It is the case of European Reference Networks (ERNs).

1.5.3. European Reference Networks

ERNs are virtual networks formed by health care providers and hospitals in Europe, with the objective of better handling rare or complex diseases, as these conditions often demand a high and specific knowledge about the case [14]. Usually, doctors from a single health centre find difficulties when dealing with this kind of patients, so they ask for an opinion to other hospitals. Then, their professionals are able to advise and help at diagnosing and planning the treatment for that situation.

Rare Diseases (RD), as the name indicates, might suggest low prevalence among people in Europe. However, there are between 5,000 to 8,000 rare conditions present, affecting approximately 30 million people inside the EU [14]. Hence, many ERNs are needed, currently existing as much as 24, which were created in 2017. Each one of them is dedicated to a specific condition.

1.5.3.1. TransplantChild

TransplantChild is the ERN on PT, led and coordinated by La Paz University Hospital. It encompasses both SOT, including multiorgan and complex procedures, and HSCT. Its main goal is to have a significant impact on children's quality of life in the long term. To achieve it, various methods are implemented. The first one consists on giving the patients network access to diagnostic advice. The second is based on joining with members of the network and stakeholders to increase the information, innovate and expertise in the transplant procedures [15].

The organization is integrated by 18 healthcare providers coming from 11 different European countries, as shown in Fig 1.5. To be accepted in the network, the centres must prove three different qualifications: enough experience in PT, a well-prepared multi-professional team, and good equipment, resources and facilities to deal with the complex procedures that may be necessary in transplantation [16]. By applying these filters at the time of accepting new members, it is guaranteed that the ERN is formed by expert hospitals only, increasing the quality of advice whenever it is needed, and therefore optimizing the treatment.

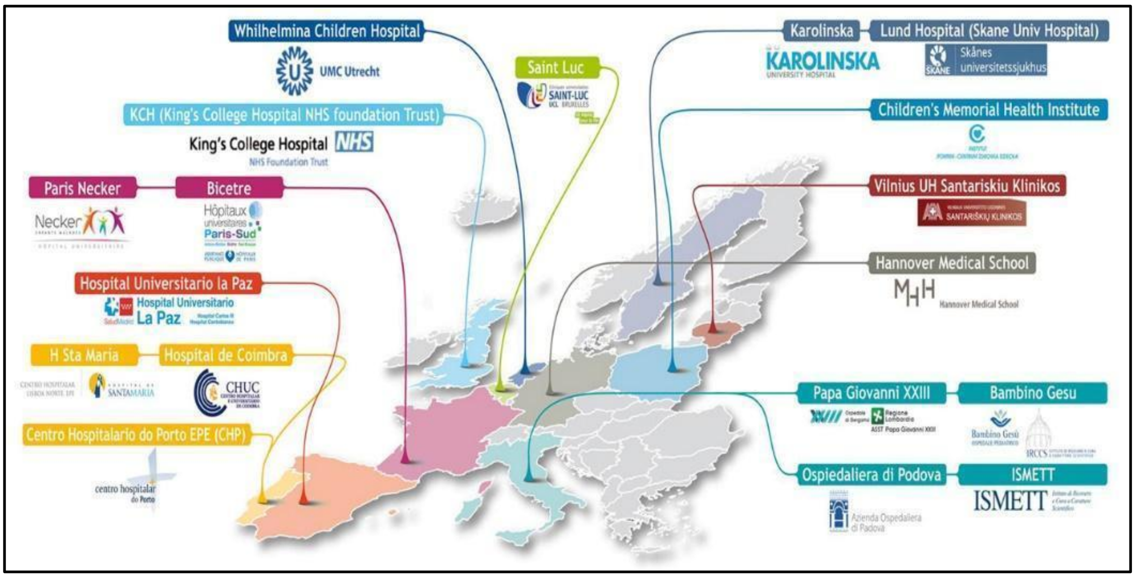


Fig 1.5. Healthcare centres members of the ERN TransplantChild [16].

1.5.4. PaEdiatric Transplantation European Registry

TransplantChild is developing an undertaking called PETER, consisting in the implementation of a registry in all the hospitals in the network, with the purpose of analysing paediatric transplants data in the EU. This project is built using a crosscutting approach, meaning that all the variables are related and analysed, independently on the type of transplant. By applying this method, it is expected to find correlations between variables from different phases of the transplantation process (pre-transplant, transplant and post-transplant). The chosen approach was to construct a registry that could reach as many patients as possible, to get consistent results [17].

As the registry is thought to be put in practice in all the hospitals belonging to TransplantChild, it is fundamental for it to be interoperable: every centre must register their patients in the same way so that there are no differences in the clinical data collected [17]. Otherwise, the information integration would not be feasible.

Moreover, in this registry it is necessary to track the patients' post-transplant information. This is achieved via different follow-ups (either visiting the hospital or electronically through the web), needed because long term data are relevant when analysing the success of a transplant. However, it is important to guarantee that patients' information is always protected. The European Union Committee of Experts on Rare Diseases (EUCERD) has stated a set of guidelines and recommendations for patient registration and data collection.

The objective behind this is helping the different institutions to build registries according to the necessary features, which are [18]:

- Being as interoperable as possible, with consistent exchange and collection of the information to achieve statistically relevant data.
- Getting information from different data sources as easily as possible. For example, from electronic records filled up by both patients and doctors.
- Utilizing obtained data for research purposes exclusively.
- Being able to adapt to regulations and requirements.
- Having a sustainable strategy for the previewed working period.

Even though this project contributed to the registry in most of its features, the main role was to develop a platform to perform the data analysis and exploitation. This application needed to follow the requisites of being:

- Integrated with the already existing platform.
- Able to perform data export, visualization and statistics.
- Available online for every hospital in the network and their personnel.
- Compliant about information security and interoperability.
- As easy-to-use as possible, so that none of the health professionals encountered any problems when taking advantage of it.

1.6. State of the art

Quality of transplants is improving in the last years. They are getting to be more common when trying to deal with complicated, and frequently rare, diseases. Despite this, performing a transplant is still a complicated process to be carried out. Therefore, there are many ways in which transplantation, as a technique, could be improved.

One of the possible directions to progress in transplantation is making new types of transplants viable. Head transplantation, for example, has caught the attention of surgeons and scientists in the last years. Some of its issues have already been solved. As a case in point, vessel anastomosis was overcome by implementing a cross-circulation protocol that prevented brain ischemia, and regarding immunosuppression, a significant progress has been made despite its pending safety and efficacy optimization [19].

Nevertheless, there are still many challenges that need a solution for a head transplant to be technically possible. Knowledge about spinal cord transection and anastomosis is low due to the difficulty of understanding propriospinal circuitry, the duration of cerebral ischemia during the surgery may be too long and spinal cord fusion is an unknown too [19].

On the other hand, the current viable transplant types need to be perfectioned. It is the case of multivisceral transplantation which, despite having significantly improved its outcomes since the introduction of tacrolimus, is still the least known type of procedure [9]. Issues such as allograft rejection, immunological monitoring and long-term survival are some of the remaining obstacles regarding this type of transplantation [9].

However, the problem of maintenance immunosuppression is not specific to multivisceral transplants, but common to all of them, as it is one of the most important aspects needing to be refined. Determining an ideal dose of immunosuppressants is crucial, since not enough quantity can lead to transplant rejection or infection, but an excessively high dosage may produce carcinogenesis or other prejudicial effects, like weakening the immune system [3].

From these ideas it can be concluded that research is fundamental in terms of improving the transplantation procedure. To facilitate investigation, data collection and analysis are necessary. These aspects are being properly developed when talking about adult transplantation, but there is a lack of PT data availability and research due to its low prevalence. This thesis is focused on solving the previous problem by implementing various tools to collect and analyse PT activity data in the EU.

Nevertheless, considering that a transplant is not just a surgery, PT activity collection is not enough to analyse the current state of the treatment. Extra data must be analysed, such as patient and graft survival: if many transplants are performed but survival rates are low, it should not be concluded that PT is in a good state.

That is the reason why follow-ups are needed, so that every transplanted patient’s data are collected and analysed, with the purpose of localizing any possible post-transplant complications and treat them in consequence. Hence, transplantation registries are necessary, as they are the best way of storing these specific types of information.

It results important to keep track of the registries that are currently available in Europe, in order to identify the needs for new ones that, instead of overlapping, complement the existing ones, increasing knowledge. In order to achieve this objective, a collection of available registries in the EU was made and organized in Table 1.2.

TABLE 1.2. COLLECTION OF AVAILABLE TRANSPLANT REGISTRIES IN EUROPE.

Registry name	Paediatric	Type of transplant	Type of data				
			Clinical	Socioeconomic	Personal	Life expectancy	Life quality
The Registry of the ISHLT (Cardiac)	Yes	Heart	Yes	No	Yes	Yes	Yes
Hemato-poietic Cell Transplantation in Japan	Yes	HSCT	Yes	No	No	Yes	No
ERA-EDTA Registry	Yes	Kidney	Yes	No	No	No	No
UK Renal Registry	Yes	Kidney	Yes	No	Yes	No	No
European Liver Transplant Registry	Yes	Liver	Yes	No	No	Yes	No
The Registry of the ISHLT (Lung)	Yes	Lung	Yes	No	Yes	Yes	Yes
The Spanish Lung Transplant Registry	Yes	Lung	Yes	No	Yes	Yes	No
Annual Report on Intestine Transplant	Yes	Multivisceral	Yes	No	Yes	Yes	No
Intestinal Transplant Registry Report	Yes	Multivisceral	Yes	No	No	Yes	No
Cancer after paediatric transplanta-tion	Yes	Solid organ	Yes	No	Yes	No	No
Eurotrans-plant	Yes	Solid organ	Yes	Yes	No	No	No

IRODaT Paediatric Transplants	Yes	Solid organ	Yes	No	No	No	No
Organ donation transplant report	Yes	Solid organ	Yes	No	No	Yes	No
Spanish Heart Transplant Registry	No	Heart	Yes	No	Yes	Yes	No
Polish HSCT Registry	No	HSCT	Yes	No	No	No	No
Registro de donantes de médula ósea	No	HSCT	Yes	Yes	No	No	No
Croatian Registry of Renal Replacement Therapy	No	Kidney	Yes	No	No	Yes	No
Estonian Society of Nephrology	No	Kidney	Yes	No	No	No	No
Finnish Registry for Kidney Diseases	No	Kidney	Yes	No	No	No	No
Registro Comunidad Valenciana	No	Kidney	Yes	No	No	No	No
Scottish Renal Registry	No	Kidney	Yes	No	No	Yes	No
SICATA	No	Kidney	Yes	No	Yes	No	No
Domino Liver Transplant Registry	No	Liver	Yes	No	No	No	No
Evolution of Liver Transplantation	No	Liver	Yes	No	No	Yes	No
The Nordic Liver Transplant Registry	No	Liver	Yes	Yes	No	Yes	No

Intestinal Transplantation	No	Multivisceral	No	Yes	No	No	Yes
Cancer Risk in Solid Organ Transplant	No	Solid organ	No	No	Yes	No	No
Collaborative Transplant Study	No	Solid organ	No	No	No	Yes	No
Section Belgian Transplant-coordinators	No	Solid organ	Yes	No	No	No	No
Status of paediatric transplantation	No	Solid organ	Yes	Yes	No	Yes	No

Source: Collected registries from PubMed and Web of Science.

It is remarkable that, even though this compilation consisted on finding registries containing specific paediatric information, only thirteen of them present some paediatric data, demonstrating the current lack of PT registration and research in the EU.

2. MATERIALS & METHODS

2.1. Analysis of paediatric transplantation activity in the European Union

Although transplantation activity is registered in the newsletter written by the Council of Europe, more details were required to perform the deep analysis proposed in this thesis. Therefore, before starting analysing the data, it was necessary to collect as much information as possible from the European countries.

2.1.1. Activity data collection

The chosen method to get the data was contacting the different transplant organizations or hospitals in every country. The proposed study was kindly explained, asking them to collaborate by sharing their activity data to the ERN TransplantChild. For instance, in Spain, the Organización Nacional de Trasplantes (ONT) was contacted via email, obtaining a fast and positive answer. Nevertheless, various messages were sent to the non-answering countries during several months, giving them more opportunities to contribute in the project.

The information was requested in a determined format, so that each country could send it the same way, facilitating its integration. Sent data had to be structured according to these instructions:

- Number of transplants divided by hospitals.
- Then, per hospital, separated by age: one group for patients younger than 16 years old, and a second one for those between 16 and 18, as patients older than 18 were not considered paediatric.
- After this, the information needed to be split according to the type of transplant: HSCT, liver, kidney, heart, lung and multivisceral.
- The next step was to cut it up by years, from 2012 to 2016.
- To finish, there was a last partition varying depending on the organ. For liver and kidney, it had to be severed by living or cadaveric (deceased) donor and, in the case of HSCTs, by autologous or allogeneic.

Despite these specific indications, many organizations sent their data incorrectly or partly missing. The countries that did (or did not) send the required information, and if they did it correctly or not, were registered as in Table 2.1. This information was stored to keep track of all the defects present in the collected data.

TABLE 2.1. REPORTED DATA FROM THE DIFFERENT COUNTRIES.

Country	Data sent	Comment
Austria	Incorrectly	Missing HSCTs
Belgium	Incorrectly	Missing HSCTs
Bulgaria	Correctly	
Croatia	Incorrectly	Missing HSCTs
Cyprus	Not sent	
Czech Republic	Correctly	
Denmark	Incorrectly	Not separated neither by hospitals nor by living/cadaveric donor Missing HSCTs Heart, lung and multivisceral were sent as a single datum for the Scandinavian countries (Denmark, Finland and Sweden)
Estonia	Incorrectly	Missing HSCTs because the responsible was not available
Finland	Incorrectly	Not separated neither by hospitals nor by living/cadaveric donor Missing HSCTs Heart, lung and multivisceral were sent as a single datum for the Scandinavian countries (Denmark, Finland and Sweden)
France	Incorrectly	Not separated by age Missing HSCTs Missing from 2012 to 2015 except for liver
Germany	Incorrectly	Missing HSCTs
Greece	Incorrectly	Missing HSCTs
Hungary	Incorrectly	Missing HSCTs
Ireland	Incorrectly	Only kidney reported
Italy	Incorrectly	Not separated neither by living/cadaveric nor by autologous/allogeneic
Latvia	Not sent	
Lithuania	Correctly	
Luxembourg	Not sent	Does not have PT activity
Malta	Not sent	Does not have PT activity
Netherlands	Incorrectly	Missing HSCTs
Poland	Incorrectly	Not separated by age
Portugal	Correctly	
Romania	Incorrectly	Only HSCTs, undivided by hospitals.
Slovakia	Incorrectly	Missing HSCTs
Slovenia	Incorrectly	Missing HSCTs
Spain	Correctly	
Sweden	Incorrectly	Not separated neither by hospitals nor by living/cadaveric donor Missing HSCTs Heart, lung and multivisceral were sent as a single datum for the Scandinavian countries (Denmark, Finland and Sweden)
United Kingdom	Incorrectly	Missing HSCTs

Source: Received data from national transplant organizations or hospitals.

As many of the organizations did not send their HSCTs information, an alternative source was found in the European society for Blood and Marrow Transplantation (EBMT) website, which registers all HSCTs in the EU per hospitals. So, in the countries with this part missing, the EBMT data was utilized. However, the only available information in this source was the total from 2012 to 2016. This lack of separation complicated the study, as it prevented the analysis per year, at least, for HSCTs.

The last consulted source was the TransplantChild website [20], that contains the activity of the network. From here, it was possible to obtain the data of both HSCTs and SOTs performed by all the TransplantChild's hospitals from 2013 to 2015. This compensated some of the information that was missing because of the incompleteness of the received activity numbers.

With the purpose of completing the obtained numbers, the transplantation newsletter from the Council of Europe was checked. It resulted useful to collect some of the missing data: the heart, lung and multivisceral transplants from Scandinavian countries were picked up from this source. As they sent this information as a single number, using the newsletter the datum for each organ was separated for Denmark, Finland and Sweden.

Nevertheless, although the information contained on this newsletter was exhaustive, it presented two different weaknesses for the purposes of this undertaking: not containing paediatric-specific HSCT data and being divided by countries instead of by hospitals (therefore, the data from the Scandinavian organizations was not collected by health centres). Because of this, the data verification was limited to the countries' SOTs sums per transplant and year, performed by comparing the number of transplants obtained for every year.

In Table 2.2, the comparison between the collected data and the ones in the transplantation newsletter is stated. For a more quantitative comparison, both the absolute and relative error were calculated when possible: absolute error was left blank if no transplantation activity was registered in none of the sources, and relative error was not computed when it resulted mathematically impossible.

Then, to help the visualization, different colours (green, yellow or red) were assigned to the data containing cells. If the two numbers coincided, the green colour was used. When the datum collected resulted to be lower than the newsletter one, the cell was coloured in red, indicating that it was incorrect: there was some missing information.

On the contrary, the yellow cells represent an increase from the number in the newsletter to the compiled one. This did not necessarily mean an error, since the newsletter paediatric age limit was 15 instead of 18. Hence, it was infeasible to know if these numbers were correct or not.

TABLE 2.2. KIDNEY COMPARISON TABLE

	KIDNEY									
	2012		2013		2014		2015		2016	
	Abs.	Rel.	Abs.	Rel.	Abs.	Rel.	Abs.	Rel.	Abs.	Rel.
Austria	6	100%	1	25%	2	29%	4	40%	3	43%
Belgium	2	33%	0	0%	5	38%	3	21%	2	13%
Bulgaria										
Croatia	0	0%	4		2		2		3	
Cyprus	-1	-100%	-1	-100%	-1	-100%				
Czech Republic	1	10%	2	22%	3	100%	-2	-22%	1	17%
Denmark	4	133%	3	30%	0	0%	4	100%	4	57%
Estonia	0	0%					0	0%	0	0%
Finland	2	33%	1	10%	3	25%	2	33%	4	40%
France	-47	-100%	-60	-82%	-42	-72%	-54	-81%	40	47%
Denmark	23	34%	18	27%	19	23%	23	32%	21	33%
Greece	0	0%	0	0%	6	100%	-2	-50%	3	100%
Hungary	-8	-62%	-3	-25%	0	0%	0	0%	6	60%
Ireland	18		0	0%	0	0%	3	60%	5	71%
Italy	6	11%	21	48%	39	108%	16	38%	44	75%
Latvia	-2	-100%	-2	-100%	-1	-100%	-1	-100%		
Lithuania	2	100%	-1	-33%	1	100%	0	0%	2	67%
Luxembourg										
Malta										
Netherlands	5	19%	3	20%	9	47%	8	53%	5	25%
Poland	-1	-2%	-7	-17%	0	0%	-2	-6%	0	0%
Portugal	6	55%	4	27%	1	17%	9	69%	2	15%
Romania	-6	-100%	-5	-100%	-1	-100%	-6	-100%	-2	-100%
Slovakia	0	0%	2	100%	2	100%	1	50%	1	50%
Slovenia	0	0%	1				1	100%	0	0%
Spain	2	3%	12	18%	4	7%	31	62%	10	19%
Sweden	2	10%	9	60%	8	67%	14	108%	8	62%
United Kingdom	36	38%	36	39%	36	37%	34	35%	36	33%

Source: Collected data and transplant newsletter [21].

Some of the imperfections of the data collected can be observed in the table. For instance, it can be seen that France presents large absolute errors and relative errors of 100% until 2015. This is because of the fact that the French institution responsible of providing the data, only sent the information regarding 2016 for all SOTs except for liver.

After the collection and verification, the information was stored in a unique Microsoft Excel file. Then, this archive was used to export the data to the different applications utilized for their exploitation and analysis.

2.1.2. Activity data exploitation

2.1.2.1. Paediatric transplants in the European Union map

The first part of the analysis consisted on building a map containing PT activity in the EU. It was decided that this task could be easily implemented while obtaining satisfactory result by using Google My Maps.

Google My Maps is a free tool useful to create customized maps. It is easy to use, as several template maps are given (political, satellite, terrain and landmass). It results very useful since the maps created in this application are saved in Google Drive, so that they are stored in the cloud and can be shared with more people if needed. Moreover, the creator can decide if the added people should be able to modify the map or, otherwise, to only visualize it [22].

The created maps can be personalized by adding new layers. They are imported from an archive in any of the following formats: CSV, XLSX, KML or GPX. These are table-like files in which each row corresponds to an element, and every column represents a feature for that element, as represented in Table 2.3.

TABLE 2.3. EXAMPLE OF BEST SUITED TEMPLATE FOR GOOGLE MY MAPS.

Country	Hospital	Address	2012	2013	2014	2015	2016	Total
Spain	La Paz University Hospital	Paseo de la Castellana, 261, 28046 Madrid	90	61	72	88	76	387

Source: Received data from the Organización Nacional de Trasplantes (Spain).

In order to locate every row, it is necessary to specify an address or a physical place recognizable for Google Maps, and then select the column that acts as address in the file. In this case, the variables needed are the country and hospital name, the mentioned address, the number of transplants from 2012 to 2016 and their sum.

The selected format for the data importation in Google My Maps was XLSX, which is obtained using Microsoft Excel. So, the statistics sent by every country were combined in a single file. However, a different archive was necessary for every object in the map. As transplants are divided in six types for this job, seven objects were imported: one for each solid organ, one for HSCTs, and one for the totals. Therefore, the main file was divided in seven, with each of them containing the information required for each layer. Besides, data per year was displayed in every layer, apart from the number corresponding to the whole-time span.

When the countries were asked to provide their PT activity information, it was requested to be divided by age because the objective was to elaborate three different maps, having the exact same structure but being different depending on their age limits: one for 0 to 16 years old, another one for 17 and 18, and a last one for the total. The three maps were built. However, due to the limitations of the data received, all the focus was put on the total map because some of the countries did not divide their data according to the proposed age limits.

The objective of working with this tool was locating all the hospitals in the map, with their respective PT activity. It is a very visual method that allows the geographical analysis of the centre and activity distribution (by number of centres and transplants), making it possible to identify patterns and defects in the hospital distribution that could be detrimental for patients living in determined areas.

Despite the many advantages provided by Google My Maps, it also had many weaknesses regarding the project’s objectives. For example, it can only display data per hospitals, and there are no more options to visualize the information. Therefore, more tools were required to complement the study.

2.1.2.2. Paediatric transplants in the European Union reports

Google Data Studio is a free online application useful to generate “interactive dashboards and beautiful reports that inspire smarter business decisions” [23]. It offers a wide variety of data formats to import and export data, as well as several ways to visualize them. Moreover, similarly to Google My maps, the work is saved in the cloud in Google Drive, so that everything can be shared, setting the privileges depending on the user.

Even though this application admits several formats to import the information, the one that was best suited for the situation was using Google Spreadsheets. This was chosen over Microsoft Excel due to two main reasons: the possibility of auto-updating the generated reports whenever new data were introduced in the online spreadsheets and the continuous online availability of the data.

The idea was to recycle all the files generated to create the previous map and import them in Google Spreadsheets so that they could act as data sources for this tool too. However, it was necessary to adapt everything to the platform's conditions.

In Google Data Studio, it was possible to display more information characteristics. Therefore, some extra fields were included in the spreadsheet, with the aim of helping data classification. The structure of the new sheet can be appreciated in Table 2.4, including the new columns added:

- Code: A hospital-specific code for their proper identification. It consisted on the ISO 3166-1 Country Codes[2] followed by a number corresponding to the hospital position in terms of number of transplants in that country.
- ERN: This variable indicates if the hospital in that row belongs to the ERN TransplantChild or not.
- Type: It classifies the row as HSCTs or SOTs.

TABLE 2.4. EXAMPLE OF BEST SUITED TEMPLATE FOR GOOGLE DATA STUDIO.

Country	Code	Hospital	ERN	Type	Transplant	Year	Number
Spain	ES01	La Paz University Hospital	Yes	Solid Organ	Heart	2016	5

Source: Collected data.

It resulted to be important for this application to have the data structured in columns, as that is the way Google Data Studio identifies variables. Hence, having each aspect in a specific column facilitated the possibility of including filters in the reports, which were important to increase their interactivity and make them able to display different information depending on the user's interest.

With this data structure, it was already possible to execute a consistent visualization. Nevertheless, in order to improve the analysis and obtain more relevant conclusions, extra variables were added for this study. The reason behind the addition of new study fields was the elaboration of a hospital's classification criteria. For that, two main characteristics were considered: activity and complexity.

---

[2] Standard method that designates countries with, in this case, two letter codes [60].

To classify hospitals' activity, their number of total transplants performed (all organs, all years) was calculated using a simple *SUM* function that selected the previous 30 rows (as there were 5 years and 6 types of transplants, each of them in an independent line). However, for hospitals with HSCT data belonging to the EBMT, as they were not per year, there was an extra row with the "All years" attribute instead of the year of transplant. Because of this, it was necessary to add an *IF* function that checked it that row was present so that, if it was, the previous 31 rows were considered instead of the 30.

Then, in a separate sheet inside the same file, the transplants per hospital average (μ) and standard deviation (σ) were calculated by using, respectively, the commands *AVERAGE* and *STDEV.S.* These quantities were employed to make an activity classification. First, each hospital's distance (d) to μ was obtained in terms of σ, by applying Equation (2.1) , where n represents the number of transplants in the hospital.

$$d = \frac{n - \mu}{\sigma} \tag{2.1}$$

Thus, the hospitals were assigned an activity score (from 0 to 6) depending on their value of d, in which the hospitals with a score of 6 satisfied that d was higher than 2.5. This means that their distance to the average was, at least, 2.5 times σ. The rest of categories were established by lowering the threshold in 0.5 for each score, in such a way that the centres with a score of 0 have a negative distance, meaning that they fall below the average.

On the other hand, a set of criteria was determined to measure the transplantation complexity of a centre and obtain a similar scale of 0 to 6:

0. If the hospital had not performed any transplant.
1. If only one type of SOT had been carried out.
2. For hospitals with just HSCTs.
3. For two types of SOT.
4. If the centre had made both HSCTs and a single type of SOT.
5. For hospitals with multivisceral transplants.
6. If HSCTs and more two or more SOT types had been performed.

2.2. PaEdiatric Transplantation European Registry

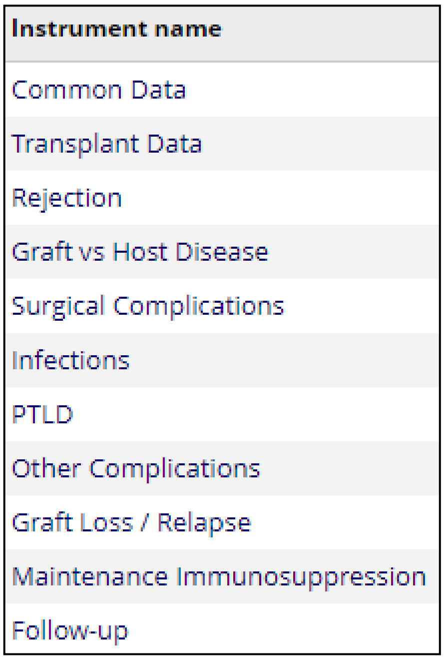
The aim of this section was to help implementing PETER, the registry proposed by the ERN TransplantChild. For that, it was necessary to develop a data exploitation platform that could get integrated with REDCap, which was the application chosen to perform the data collection.

2.2.1. Patients' data collection: REDCap

"REDCap (Research Electronic Data Capture) is a secure, web-based application [9]designed to support data capture for research studies, providing 1) an intuitive interface for validated data entry; 2) audit trails for tracking data manipulation and export procedures; 3) automated export procedures for seamless data downloads to common statistical packages; and 4) procedures for importing data from external sources" [24].

TransplantChild requested a REDCap license to implement this registry. However, even though REDCap is free, it is limited to institutions and organizations, so the agreement was rejected (since TransplantChild is an ERN, not an institution). Because of this, the study was carried out as a specific project inside the REDCap server hosted at IdiPAZ (Hospital La Paz Institute for Health Research).

Some registry features were already implemented in the REDCap project as part of a previous Bachelor Thesis [12], so this project's goal was to strengthen some aspects that were unfinished and investigate extra features from REDCap that could improve its functionality for the desired purposes.



Instrument name
Common Data
Transplant Data
Rejection
Graft vs Host Disease
Surgical Complications
Infections
PTLD
Other Complications
Graft Loss / Relapse
Maintenance Immunosuppression
Follow-up

Fig 2.1. REDCap data collection instruments applied for PETER [12].

The forms for the data entry consisted on complete questionnaires including several aspects from pre-transplantation, transplantation and post-transplantation. The focus was put on the latter information, and especially on the possible complications that could affect the transplanted patients.

Data were divided in eleven types of instruments as indicated in Fig 2.1, in order to be able to separate different types of information that may be exploded independently. Some of them are repeatable, meaning that more than one survey can be filled for that patient and instrument. For instance, if a patient has experienced two different rejections, this instrument is duplicated, so that each rejection is registered independently. To provide an example, the fields used in Rejection and their formats are indicated in Table 2.5.

TABLE 2.5. REDCAP REJECTION FIELDS.

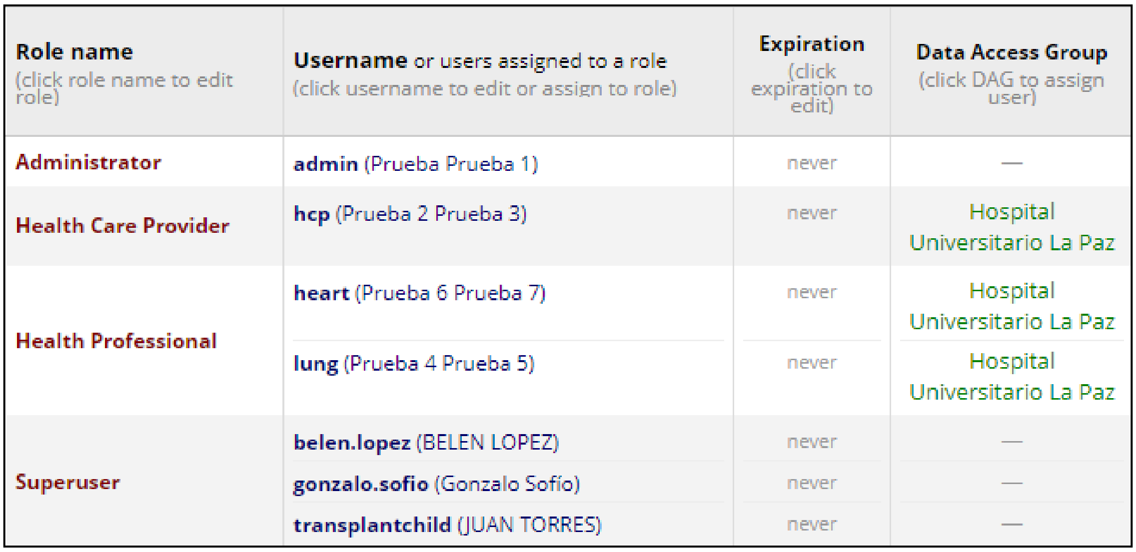
COMPLICATIONS: REJECTION			
Post Transplantation: Rejection			
Field name	**Variable name**	**Section**	**Choice**
Rejection?	reject	Transplant data	0, Yes | 1, No
Biopsy proven?	biopsy_reject	Transplant data	0, Yes | 1, No
Date of Rejection	reject_date	Transplant data	dd/mm/yyyy
Cause of rejection	cause_reject	Transplant data	(Free text)
Type of rejection	typereject	Transplant data	1, Hyperacute | 2, Acute | 3, Chronic
Mechanism of rejection	reject_mech	Transplant data	1, Cellular | 2, Humoral
Rejection description: Histological	reject_histo	Transplant data	(Free text)
Rejection description: Clinical	reject_clinic	Transplant data	(Free text)
Rejection Treatment Description	reject_treat	Transplant data	1, Corticosensible | 2, Corticoresistant | 3, Other
Immunosuppressive Rejection Treatment	im_reject_treat	Transplant data	1, Cliclosporin | 2, Tacrolimus | 3, Sirolimus | 4, Microphenolic Acid | 5, Corticosteroid | 6, Other
Biological Treatment	bio_treatment	Transplant data	1, Thymoglobulin | 2, Alemtuzumab | 3, Basiliximab | 4, Rituximab | 5, Other
Other Rejection Treatment Description	oth_reject_treat	Transplant data	(Free text)
Outcome	outcome_reject	Transplant data	1, Resolution | 2, Graft Dysfunction/ Partial Graft Loss | 3, Complete Graft Loss

Source: Implementation Tools for Data Integration and Knowledge Sharing in the European Reference Network: Transplantchild [12].

Moreover, the Follow-up instrument is collected every time it is scheduled: in this registry, the follow-ups are thought to be scheduled for one, three, five and ten years from the transplant.

2.2.1.1. User roles & Data Access Groups

The platform was conceived to be available for the personnel working in the registry. Therefore, it was necessary to establish different user roles as in Fig 2.2, by defining the privileges of each worker when treating the data:



Role name (click role name to edit role)	**Username** or users assigned to a role (click username to edit or assign to role)	**Expiration** (click expiration to edit)	**Data Access Group** (click DAG to assign user)
Administrator	**admin** (Prueba Prueba 1)	never	—
Health Care Provider	**hcp** (Prueba 2 Prueba 3)	never	Hospital Universitario La Paz
Health Professional	**heart** (Prueba 6 Prueba 7)	never	Hospital Universitario La Paz
	lung (Prueba 4 Prueba 5)	never	Hospital Universitario La Paz
Superuser	**belen.lopez** (BELEN LOPEZ)	never	—
	gonzalo.sofio (Gonzalo Sofío)	never	—
	transplantchild (JUAN TORRES)	never	—

Fig 2.2. REDCap user roles and Data Access Groups.

- <u>Superusers</u>[3]: Their function is to administrate and regulate the project. They have all the privileges and can access all the information.
- <u>Coordinators/Administrators</u>: They act as supervisors. They have all the information privileges, but they cannot change the platform in any way. For example, they cannot edit the user rights neither the project design.
- <u>Health Care Providers (HCP)</u>: They are responsible of coordinating the Health Professionals from their respective hospitals. This is achieved by using Data Access Groups (DAGs). Their rights are the same as the Coordinators’, but their

---

[3] These are user roles regarding the specific REDCap project which, in this case, was the registry. This superuser is different than the REDCap superuser which is the server and platform administrator.

information access is restricted to only the hospital where they work. One DAG is created for every hospital, to allow the restriction for every HCP.

- Health Professionals: The main difference between HCPs and Health Professionals is that these lasts can edit and modify survey responses, as they are the ones in charge of filling them up too.

2.2.1.2. Other features

Other REDCap aspects were tested to perfect the functioning of the platform. Reports and stats were generated to check if REDCap could be useful for data exploitation too, but with negative results as they were not complete enough. Also, Data quality rules were implemented, to ensure a correct information collection (required fields, missing values, numerical outliers, incorrect values for calculated fields, etc.).

The last element investigated was the Project bookmarks module: it consists on redirecting to specified webpages inside or outside REDCap. At first, this module was only utilized to link the platform to the map and the reports previously generated. Nevertheless, this function resulted to be key as it facilitated the connection with the data exploitation platform.

2.2.1.3. External modules

In REDCap, it is possible to create and implement external modules that get integrated and add extra features to the platform. Modules created are reviewed by the Vanderbilt REDCap Technical Team, to check they follow the required guidelines [25]. Some of the main recommendations are [25]:

- The module must be published on a public GitHub[4] repository with a visible open source license.

---

[4] GitHub is a social network that acts as a code-hosting repository. It makes user's information visible while presenting their open source projects [61].

- It needs to be clearly described, execute without errors and have a complete functionality.
- The maintainability should be guaranteed for both large and small projects. Also, the use of Composer[5] is recommended.
- The module must be compatible with the REDCap and PHP versions, and have the required permissions.
- REDCap authentication and user roles need to be used in the module if necessary.
- Potentially malicious code is not allowed in the module.

As stated in the guidelines, all the existing external modules are collected in a GitHub repository and are free to use, so they were explored to see if any of them could complement the existing REDCap platform.

- User Profile: This module completes user's information according to the project's needs [26]. It would be useful to identify TransplantChild users with their respective hospital, country and role details.
- Auto-Schedule: It automatically schedules events after saving records. There exist different options, like generating a schedule using a date specified in any field or from record creation [27]. With this, it would be possible to auto-schedule patients' follow-ups utilizing the date of transplant as baseline.
- Email Alerts: It allows REDCap to send emails with different types of content to determined users. They are sent when a condition is satisfied [28]. In the registry, this module would be used to remind doctors to fill up follow-ups when the date arrives.
- CSS Injector: With this extension, it is possible for administrators to insert CSS[6] into the forms [29]. It is useful to change the style of the form according to the registry design.

---

[5] Composer is a tool for managing project's dependencies in PHP, so that the libraries utilized are installed and updated automatically [62].

[6] CSS (Cascading Style Sheets) is the language that describes the presentation of HTML elements on the screen.

- Form Render Skip Logic: This module “would allow an instrument designer to choose to show or hide entire forms based on the value of a single REDCap field” [30]. Therefore, instruments corresponding to post-transplant complications could be hidden for patients that have not been transplanted yet.
- Auto-DAGs: It automatically generates and assigns DAGs to records depending on a determined field [31]. With it, patients would be added to their respective hospital DAG automatically on registration.

As the REDCap platform was hosted at IdiPAZ, it was necessary to have an IdiPAZ administrator account to be able to add the modules. This was not possible at that time, as REDCap only allows one administrator per distribution, so the external modules were saved for possible future improvements. They would be added if a REDCap owned by TransplantChild was obtained at any time.

2.2.2. Platform selection for data exploitation

One of the main works for this undertaking was the development of a platform in which data could be analysed and exploited, as the data collection one was already constructed. The first option was to use an existing software and adapt it to the project necessities, as developing a platform from scratch would demand more work and time. Hence, many options were explored with the aim of obtaining a provider or a software that could facilitate the design of the platform.

REDCap

REDCap would have been the ideal solution, as the whole registry could have been integrated in the same platform. However, although it was useful for data collection, for the exploitation, extra features were needed, like making customizable plots and calculating statistical variables. That is the reason why it was decided to test other solutions and integrate the chosen one with the existing REDCap platform.

Pathfinder

"Pathfinder is an intuitive and effective disease management system, which offers care teams a modular, plug and play solution to treat chronic disease patients across continuum of care more successfully. The solution can cover different organisational needs – to quickly build a simple disease registry or cover comprehensive care coordination requirements" [32].

The software was developed by Marand, which is a Slovenian company focused on developing new IT solutions for healthcare [33]. In Pathfinder, clinical data is stored in openEHR format[7] [32]. It is the option utilized by Eurotransplant in their registry, listed in Table 1.2.

As there was no trial or demo account for this solution, it was decided to contact Marand directly. A first online meeting was appointed with the aim of explaining the registry plan. After that, in a second web conference, the company illustrated the product features.

The conclusion was that Pathfinder could act as a REDCap substitute, as it turned out to fulfil the data collection requirements perfectly. However, it was not worth to do this substitution, since everything was already implemented in REDCap. Regarding the data analysis part, tables were the only integrated method present for data visualization, as that part of the product was still on development, not satisfying the registry needs at that time.

LabKey Server

LabKey Server is a web application that stores data in a relational database engine. It consists on data storage, file management and security as well as specialized modules that support specific scientific scenarios [34]. It is based on a security model that utilizes a complete system of users similar to REDCap, presenting groups, roles and even specific user permissions, that are enforced in every possible way of accessing the information [34].

---

[7] OpenEHR format is a set of instructions built to facilitate the flexibility and interoperability of Electronic Health Records (EHR) [63].

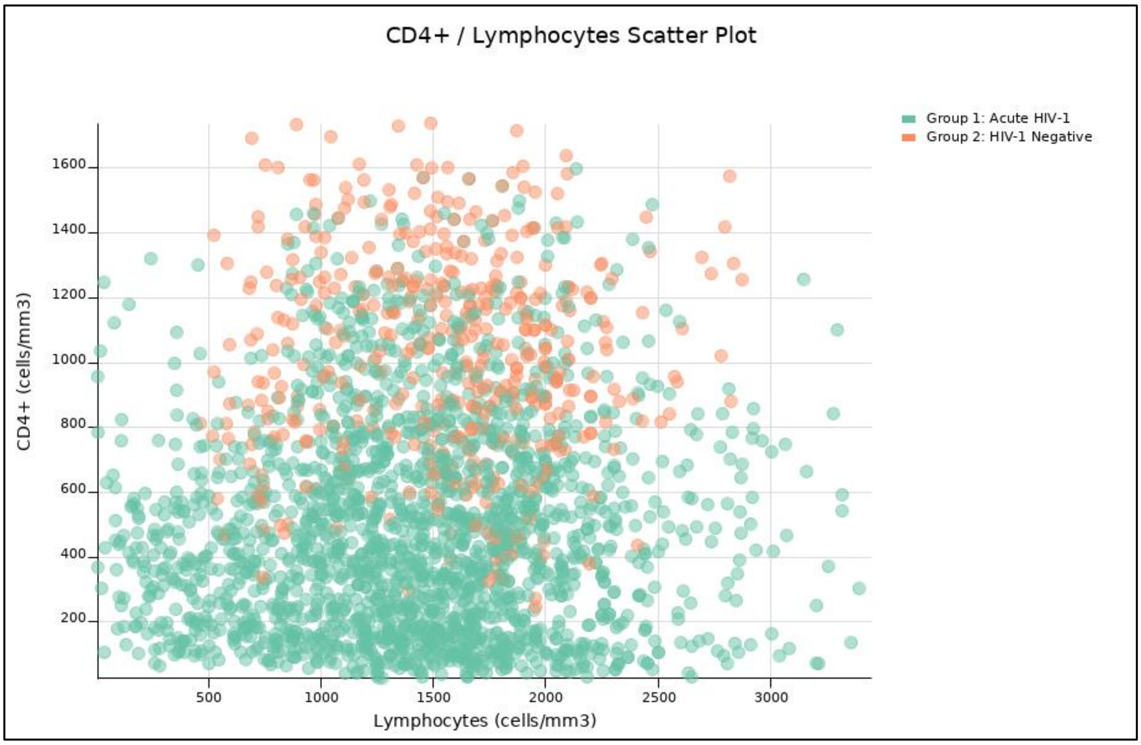


Fig 2.3. Example of a scatter plot obtained during the LabKey tutorial.

For its evaluation, all the tutorials were completed to get familiarized with the platform. The obtained results were satisfying for the registry purposes, as both the data treatment and security and exploitation were as needed. Its data visualization capabilities can be appreciated in Fig 2.3.

Still, as the data was stored in REDCap, it was necessary to connect these two platforms so that the information was sent to LabKey automatically for its exploitation. There was an existing module for LabKey to get integrated with REDCap. However, this module was not available in LabKey Community, which is the only free version. The versions that do include REDCap integration are worth, at least, 43,000$ per year, which was far beyond the budget available for the project.

JMP

The statistical software JMP created by SAS is a tool that combines strong statistics with dynamic plots. As it is interactive and visual, it is able to identify parts of knowledge that are usually difficult to observe in tables and static graphs [35].

It was thoroughly tested through its demo, and it resulted to be a strong option in terms of data analysis. It presented abundant types of plots, statistics, and the option of designing customized reports through the creation of scripts.

However, after having tried it, the conclusion was that it was not valid because it did not have the online competences expected for the registry, as it was missing user roles, REDCap integration, etc. Besides, the company was asked for a price and it was also too much for the available budget, as it was organized in individual licenses costing around 1,000$ each (depending on the necessary number of licenses), and TransplantChild needed many of them.

TranSMART

TranSMART is an effective application created by various informatics groups with the objective of integrating data from internal and external sources in the same platform [36].

It was recommended to TransplantChild in a meeting with researchers from the Health Institute Carlos III (ISCIII), with the objective of integrating clinical and genomic data. Even though genomic data was not included in the registry, it was decided to try this software and its competences.

As expected, it was based on biomarker data and none of the data types supported were compatible with the collected information formats. Moreover, the software was designed to work on Linux, which was not an option at that moment as none of the available computers had it installed.

Therefore, in order to test the software, it was required to use a virtual machine (VirtualBox, in this case), as advised in the TranSMART documentation for Windows users [37]. Although this made possible the software testing, it was not enough to be used for production mode, as warned in the website [37].

Developed software

Since the previous software did not provide any suitable solution for the platform development, it was decided to use a Python to develop it from scratch.

“Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse” [38].

There exist more than 160,000 Python projects stored in PyPI (The Python Package Index), consisting in making packages and libraries that extend and complement Python's functionality [39]. Some of the most important and widely used packages are:

- NumPy: It "is the fundamental package for scientific computing with Python" [40]. It enables new features like working with linear algebra, random numbers, arrays and matrices or dealing with complex functions [40].
- pandas: "pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language" [41].
- Matplotlib: This library is helpful to generate every kind of figures in different formats and environments [42].

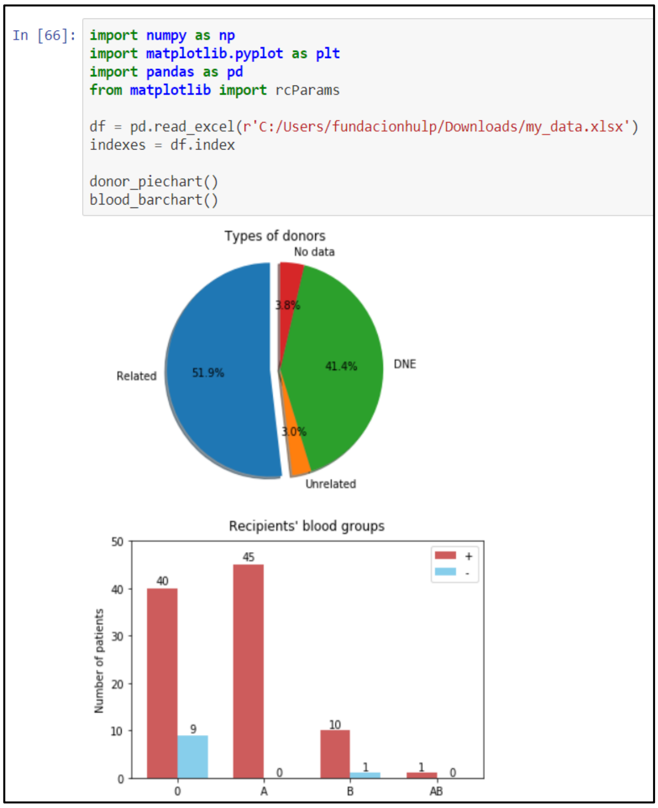
Python, as a programming language, is not a proper tool or software to develop the necessary platform. However, many applications are based on Python computing, and some of them can be suitable options for this task. In fact, two different Python-based options were explored: Jupyter Notebook and Flask.

Jupyter Notebook

"Project Jupyter is a non-profit, open-source project, born out of the IPython Project in 2014 as it evolved to support interactive data science and scientific computing across all programming languages" [43]. It consists on a set of tools for scientific computing and data science across several programming languages like Python, Julia or R. Its main application is the Jupyter Notebook, which is a web platform for programming and combining it with narrative text and other media. Therefore, a Jupyter Notebook results in a computation report that can be easily shared, visualized and understood [44].

In this software, the code is organized in cells. This facilitates the execution of pieces of code independently. However, this property can complicate everything if it is not used properly, since the order in which they are executed must be considered. For instance, if a cell that uses any command from the NumPy package is executed before the cell in which the package is imported, an error is returned. Nevertheless, if this characteristic is well-used, it can facilitate the workflow.

It was decided to test Jupyter Notebook by creating a test notebook in Python. Some sample transplant data from a different study were used to generate the graphs in the report. In Fig 2.4, the code written in the first cell of the test notebook generated is represented with its respective output. It can be appreciated that the plots' codes are not present in the figure. The cell is only importing the necessary packages and importing the sample data. Then, it calls the plots, but they must have been generated previously for this computing to work. In fact, this makes the notebook workflow non-optimal, because for its optimization it should be executed in the following order:



```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from matplotlib import rcParams

df = pd.read_excel(r'C:/Users/fundacionhulp/Downloads/my_data.xlsx')
indexes = df.index

donor_piechart()
blood_barchart()
```

Fig 2.4. First cell of the test Jupyter Notebook and its output.

- First, import the packages and read the file.
- Then, generate the graphs.
- Lastly, call the graphs for them to be displayed.

Still, as this notebook was part of testing the functionalities, this non-optimization was useful to understand its working mechanism.

The result of the evaluation was satisfying in terms of data exploitation. One of the most important advantages was the possibility of customizing the analysis by just changing the Python code. So, it was possible to generate complete and easy-understandable reports with this tool that could help analysing the patients' data. It was decided that this property compensated the fact that it had to be developed from scratch, requiring more work.

Nevertheless, Jupyter Notebook could not satisfy some of the registry necessities, like presenting an online workflow and availability or distinguishing between users with their respective privileges.

Flask

Flask is a microframework designed to create websites and web applications using Python. It relies on three subsystems that are Werkzeug, Jinja2 and Click [45]. “There is no native support in Flask for accessing databases, validating web forms, authenticating users, or other high-level tasks. These and many other key services most web applications need are available through extensions that integrate with the core packages” [45].

Flask and Jupyter Notebook present similarities, as both of them are based on Python programming. Therefore, Flask, as well as the previous application, presented an advantage and a disadvantage when compared to the rest of platforms: it was fully customizable, but it had to be built from scratch. As previously stated, it is worth it.

On the other hand, it was very different to Jupyter Notebook regarding its online capacities: as a website, it was possible for it to be available online to everyone in the network at the same time, but user roles needed to be implemented.

2.2.2.1. Platform comparison

After having tested all the different options and collected their advantages and disadvantages regarding the registry design, it was chosen to build a web application in Flask from scratch. It resulted to be the most consistent option in terms of covering all the necessary features, with the only drawback of requiring more computational work. Every aspect considered in the platform selection is stated in Table 2.6.

TABLE 2.6. SOFTWARE BENEFITS AND DRAWBACKS.

	Benefits	Drawbacks	Comments
REDCap	-Automatically integrated with data collection. -Available online. -User roles defined. -Free.	-Too weak for data analysis.	Graphs were only available in reports, with just a few predetermined options. It improved with REDCap Chart Field module but was still weak.
Pathfinder	-Consistent data collection tool. -Possible REDCap substitute, and more customizable.	-Not enough data analysis (still on development).	Might have been a considerable option if it was fully developed.
LabKey Server	-Powerful data visualization. -Good data treatment and security. -Available online. -User roles. -Free version.	-REDCap integration only available in Pro versions. -High price.	The price exceeded the available budget.

	Benefits	Drawbacks	Comments
JMP	-Strong data analysis. -Customizable through scripts.	-Not consistent for an online registry. -Price per license: expensive for multiple people performing the analysis.	It was a great option for offline data exploitation, but not for this project.
TranSMART	-Consistent data plotting and statistics.	-Not thought for clinical data (different formats). -Focused on genomic data. -Operating System selective.	It was not adapted to the project objectives.
Jupyter Notebook	-Fully customizable data exploitation. -Possibility of connecting to REDCap.	-Not available online. -Work from scratch.	It could be embedded online, but without the option of multiple people analysing the data.
Flask	-Fully customizable data exploitation. -Possibility of connecting to REDCap. -Available online.	-Work from scratch.	It was able to cover all the needs, in exchange for more work.

Source: Tested software for the platform design.

2.2.3. Platform development

The platform development in Flask was divided in smaller tasks, with the purpose of checking that all the requirements were fulfilled and facilitating the workflow.

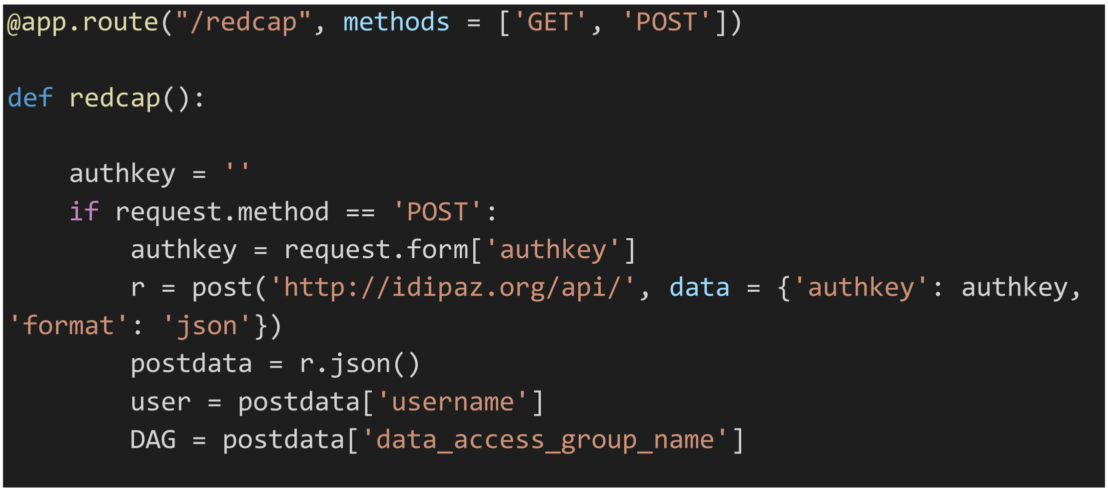
2.2.3.1. REDCap integration

Since REDCap is not an open source software, it was not feasible to modify its code, so a full platform unification was unreachable. Despite this, it was possible to integrate both platforms, in such a way that both could be accessed from REDCap and shared the data and users.

This was achieved through the Advanced Link option inside the REDCap Project bookmarks. Setting up a bookmark via an advanced link provides more features than a simple link: apart from directing the user to the desired website, it sends extra information to the latter. Several steps needed to be followed in order to use this feature properly:

- REDCap sends a variable called *authkey* to the external website using an HTTP[8] POST[9] request when the user clicks on the link.
- After this, the external web ought to send back the same key to REDCap via another POST request to the REDCap Application Programming Interface (API). The request must contain two parameters: *authkey* (the same one generated by REDCap previously) and *format*, which specifies the format of the response data, and should be either *csv*, *json* or *xml*. It needs to be sent to, in this case, the Uniform Resource Locator (URL) *http://idipaz.org/api/*.
- Then, REDCap returns, in the determined format, the following information:
  a. The username of the user who clicked on the link.
  b. The identification number and URL of the REDCap project.
  c. The name and number of the user's DAG.

So, first, the local URL *http://127.0.0.1:5000/redcap* was created to locate the web application and follow the previous steps to obtain the user's information, as in Fig 2.5.



```
@app.route("/redcap", methods = ['GET', 'POST'])

def redcap():

    authkey = ''
    if request.method == 'POST':
        authkey = request.form['authkey']
        r = post('http://idipaz.org/api/', data = {'authkey': authkey,
'format': 'json'})
        postdata = r.json()
        user = postdata['username']
        DAG = postdata['data_access_group_name']
```

This methodology already provided a high level of security to the website, making it compulsory to access it through REDCap. Nevertheless, adding extra protection is always

---

[8] HyperText Transfer Protocol (HTTP) is an application-level protocol based on request-response, in which users send a request to a server with a message containing relevant information. Then, the server responds with a status line, a success or error code, body content or many more options [64].

[9] POST is a request method designed to cover functions like annotation of resources, posting messages or providing data [64].

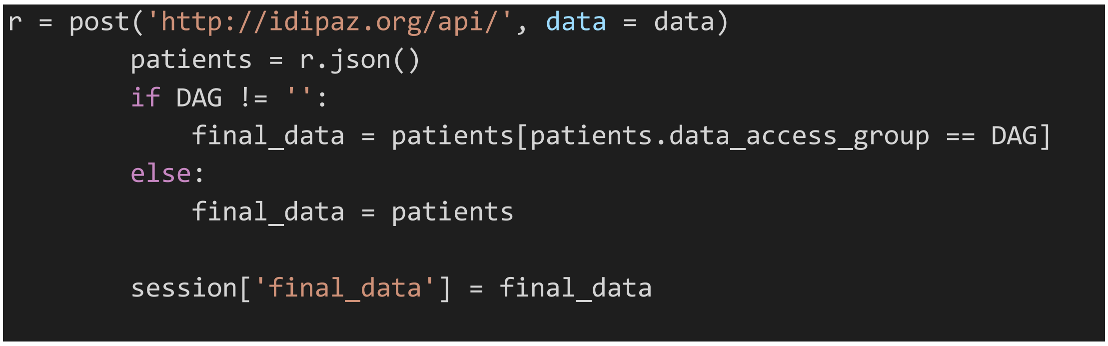
beneficial. Because of this, an *if* statement was added in which, if the two conditions are fulfilled, then the data from REDCap is exported. Otherwise, an error occurs. These conditions are useful to check if either of the *authkey* or the *user* are blank. In this way, if any of them is empty, an error is returned.

After this, a new variable *session* containing the username is created. It is useful to identify the user when navigating through the different routes in the website. In order to keep it safe, it is necessary to generate a secret key, in *config.py*, that prevents the session modification by the user. When the browser is closed, the session expires, so it is required to access through REDCap again.

2.2.3.2. Data export

The next step consisted on exporting all the patients' data to the web. In order to do so, it was necessary to make a new POST request to the REDCap API. However, in this case, it is performed differently, since it is a request to obtain the project's data specifically. For that, an API token was required, which allows the user to use a determined project's API. Then, introducing the API token as an input in the data dictionary for the request, it was possible to export and save the project records in a JSON (JavaScript Object Notation) object called *patients*.

When this is done, before providing full data access, the *DAG* must be checked so that the user can only see the allowed information. To achieve this, it must be examined if the variable is blank, which means the user has full data access. Otherwise, *patients* is converted into a different object containing only the patients with the same DAG as indicated by the code in Fig 2.6.



```
r = post('http://idipaz.org/api/', data = data)
        patients = r.json()
        if DAG != '':
            final_data = patients[patients.data_access_group == DAG]
        else:
            final_data = patients

        session['final_data'] = final_data
```

Fig 2.6. Piece of code used to check the user's DAG.

At this point, both the user's and patients' data are loaded and stored in *session*. This was all achieved inside the */redcap* URL. This route is just accessed when clicking on the

REDCap bookmark and not anymore, as the process of data export would otherwise take place again. After this, the user the user is redirected to the home section so that he can proceed to the next step: data exploitation and analysis.

2.2.3.3. Data exploitation

As a beginning, to facilitate its use and implementation, the PETER data examination was thought to be simple: focused on basic fields to give a general view of the situation in the hospitals from the ERN. Therefore, the study was constituted by the following parts:

- Plotting of the cumulative number of patients on the waiting list and transplanted in the previous year.
- Comparison of the final previous data with the ones from previous years.
- Graphical representation of graft and patient survival.
- Registration of total follow-ups made, subdivided by patient status at the follow-up.

However, as the registry is not implemented yet, patients' data are still not available and, therefore, they are not collected in REDCap either. Because of this, the visualization was illustrated by creating random but realistic data in Python for every plot. Then, the graphs were defined as functions in a different file, *plots.py*, which was later imported in the main one, *main.py*. Whenever real information is collected, the random data utilized will be substituted by the real ones.

Four different plots, represented in Fig 2.7, were generated for the complete analysis, with the possibility of filtering in each of them to make it more consistent and customizable. Nevertheless, it was not possible to implement the filters yet due to the lack of real data until the registry is in production mode.

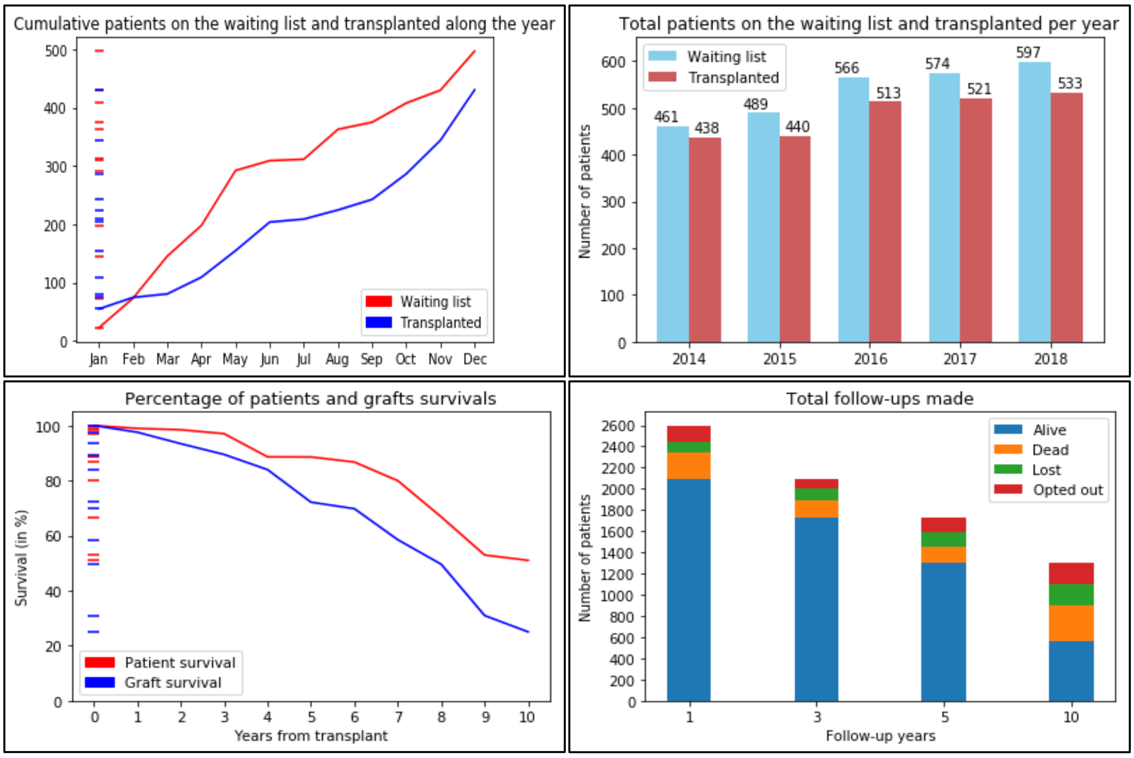


Fig 2.7. Plots to be included in the Flask web application.

Upper-left: Number of patients on the waiting list and patients transplanted in 2018. Upper-right: Historical waiting list and patients transplanted in the last 5 years. Lower-left: Patient and graft survival for all patients. Lower-right: Total follow-ups made by type.

Note: the data plotted are not real but randomly generated.

2.2.3.4. Web structure

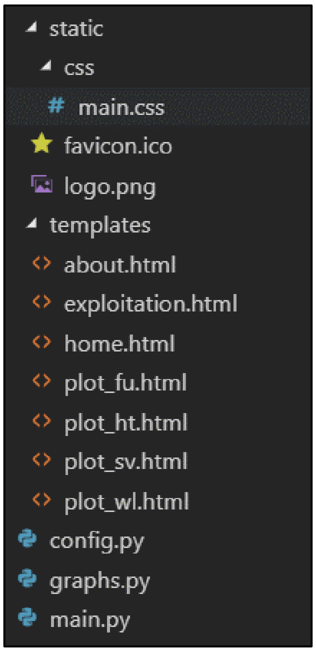
It is important for a website to have a clear structure that facilitates its use and organization. In this case, the proposed distribution consisted on the following routing:

- Home: When the link inside the REDCap project is clicked on, the user is redirected to this page, in which a brief introduction is present, and the user is given instructions to start exploiting data.
- Data exploitation: It is a set of four different directions, one for each of the plots, in which the respective plot and filters are shown for its visualization and analysis.
- About: In this section, all the information about PETER can be found, as well as the contact data if the user finds any difficulty or problem.

2.2.3.5. Web presentation

In Flask, all the actions and processes are carried out in the Python files. However, to implement the project as a web application, it is necessary to add extra files to obtain a proper executable website, following the structure specified in Fig 2.8.

The templates are *.html* files that build the structure of every route and need to be stored inside the *templates* folder. On the other hand, the static files are stored inside the *static* folder like, for example, the embedded images. Lastly, the styling is achieved by creating CSS files and placing them in a *static* subfolder called *css*.



```
static
  css
    # main.css
  favicon.ico
  logo.png
templates
  about.html
  exploitation.html
  home.html
  plot_fu.html
  plot_ht.html
  plot_sv.html
  plot_wl.html
config.py
graphs.py
main.py
```

Fig 2.8. Structure of the application folder.

In all the templates, the header is very similar. Initially, the page title is indicated, together with the links to the icon and the stylesheet used (*main*.css). The first part consists on a *h1* (header one) with the registry acronyms, an image containing the TransplantChild logo and the full registry name in *h2* (header two).

The second part of the header is formed by the navigation menu that contains the different routes. It is made with an *ul* structure with each route in a separate division. Moreover, the directions for the data exploitation are hidden inside a dropdown button, so that they are only shown whenever the cursor is hovered over this button.

Regarding the body of the page, it is different for every route. In Home, it contains an introductory text to the platform, plus two specific indications to either continue to Data exploitation or, on the other hand, go back to REDCap. It is important to separate everything using *p* and *div* structures (paragraphs and divisions), so that styling and positioning can be assigned separately.

Secondly, in the general route for Data exploitation, the different plots are explained, with links to all of them, by using different paragraphs for each description and a button for every link. At last, in the About section, a text describing the registry characteristics can be found, as well as a separate paragraph with the contact information to send suggestions or possible platform issues.

With respect to the style, many aspects needed to be determined to present the give the application a correct presentation. The TransplantChild logo was set to float to the right of the first header, while the second header was placed immediately under these two. Then, the menu was set as a horizontal bar displaying the Home and Data exploitation (with its respective hidden dropdown content) sections on the left and, on the right, About. Also, in the menu, the current active page is differentiated with a different background colour.

In Fig 2.9, the styling code for three different elements can be appreciated: the general text, the plots and the buttons. As shown in the image, each element is named using a different format. This is because *text* is a *class*[10], *plot* is an *id*[11] and *input* is a general element format. In stylesheets, a point is used to identify a *class* and a hash for an *id*.

```
.text{
  text-align:justify;
  padding: 0px 200px;
  max-width: 50%;
  position:relative;
  top:60px;
  margin-left: auto;
  margin-right: auto;
}

#plot{
  display: block;
  width: 768px;
  height: 576px;
  position: static;
  margin-left: auto;
  margin-right: auto;
  padding:50px 0px 20px;
}

input{
  background-color: #f29401;
  color: #003e89;
  border: 2px solid #003e89;
  border-radius: 10px;
  margin-bottom: 40px;
  padding: 10px;
  cursor: pointer;
  box-shadow: 0 6px 6px 0 #003e89,
              0 6px 20px 0
rgba(0,0,0,0.19);
}
```

Fig 2.9. Styling code for the general text, the plots and the buttons.

General characteristics are given for many items, such as the position, the colours, or the margins and paddings. Apart from that, specific style indications are coded for each, like the text justification, the plot size, and the pointer cursor on the button.

---

[10] "The HTML *class* attribute is used to define equal styles for elements with the same class name. So, all HTML elements with the same *class* attribute will have the same format and style".

[11] "The *id* attribute specifies a unique id for an HTML element (the value must be unique within the HTML document). The id value can be used by CSS and JavaScript to perform certain tasks for a unique element with the specified id value.

3. RESULTS

3.1. Analysis of paediatric transplantation activity in the European Union

3.1.1. Paediatric transplants in the European Union map

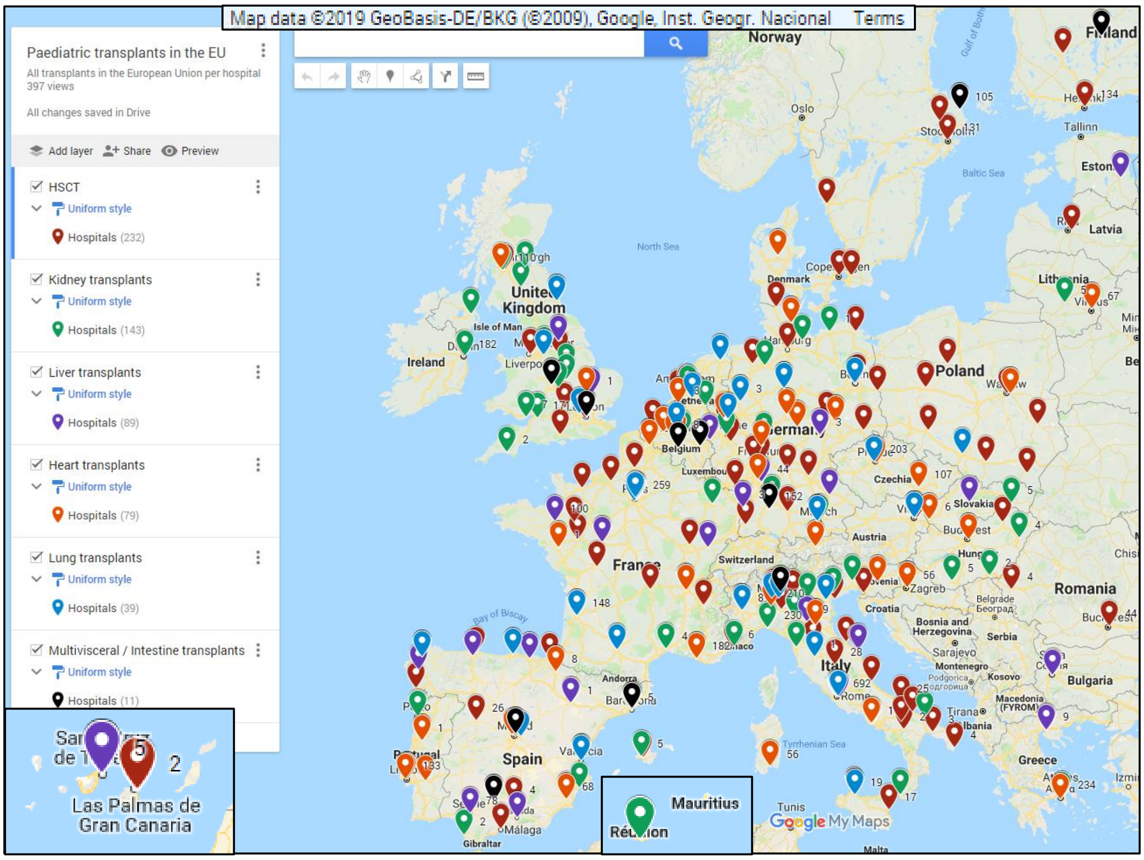


Fig 3.1. Paediatric transplants map: total activity.

In the squares are represented the Canary Islands (Spain) and Réunion (France).

As mentioned in section 2.1.2.1., three maps were generated but, as the data collection was incomplete, all the focus was put on the total map. This one integrates all paediatric transplants without any age threshold, compensating for the countries that did not divide their data by age.

In Fig 3.1, all the hospitals in the EU with any PT activity are represented. In it, all the layers are displayed at the same time. However, instead of showing it this way, it is more practical to just show the layer of totals. Otherwise, different types overlay each other, so: if a hospital has performed both HSCT and kidney transplants, that hospital appears only as kidney transplant, as that is the top layer. With the aim of minimizing this effect, the layers were put in order, from the most common type of transplant (at the bottom) to the least common (at the top), so that the less repeated transplants, like multivisceral, are not hidden. Still, it is recommended to display them one by one so that facilitates the identification of all the transplantation centres that perform a determined type of transplant, as in Fig 3.2.

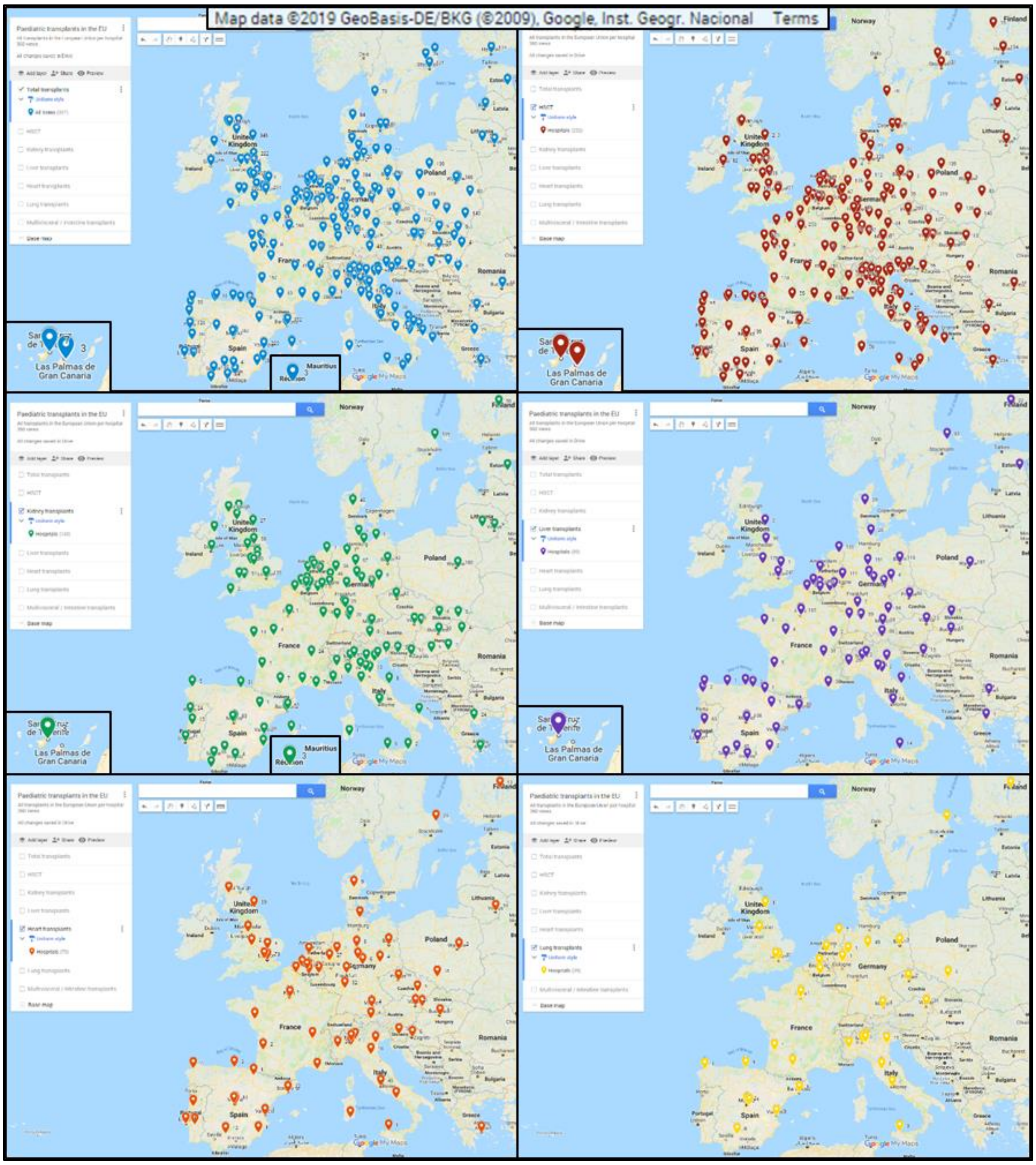


Fig 3.2. Paediatric transplants map: Different types' layers.

A) In blue, the totals. B) In dark red, HSCT. C) In green, kidney. D) In violet, liver. E) In orange, heart. F) In yellow, lung.

With this information, it is possible to identify and propose solutions for problems in the geographical distribution of hospitals. As multivisceral transplants are the least common ones, it is easier to find these issues, so the analysis was performed on this transplant type.

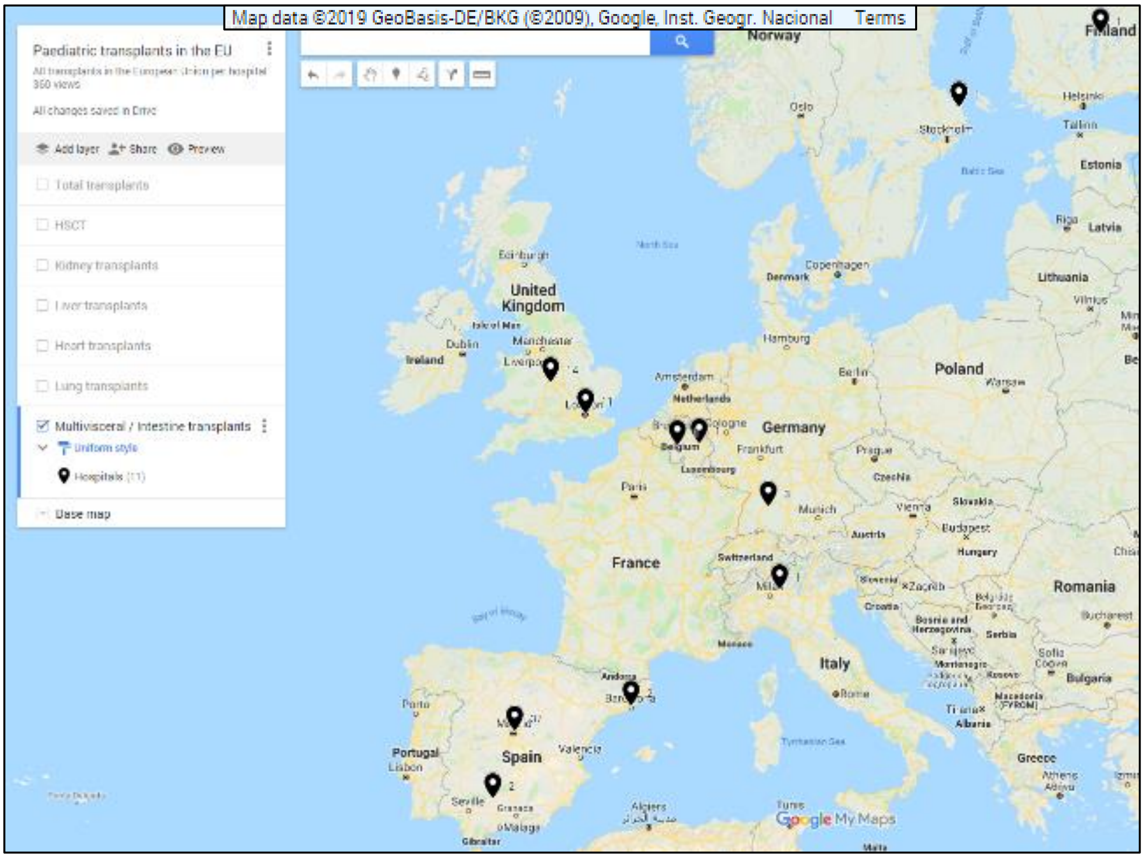


Fig 3.3. Multivisceral transplants activity in the EU.

In Fig 3.3, the low number of paediatric multivisceral transplants performed in the EU can be appreciated. When analysing possible issues, France was not considered, due to their data inaccuracy.

Only some countries present multivisceral activity: the ones coloured in blue in Fig 3.4. Therefore, international collaboration is necessary to treat all the patients with this indication. For instance, if a child living in Vienna (Austria) needs an intestinal transplant, the patient should be moved to either of the Tübingen Transplantation Centre (Germany) or the Papa Giovanni XXIII Hospital in Bergamo (Italy), as those are the closest centres with that type of transplant. However, these hospitals have only performed 3 and 1 transplants respectively so, if the case is especially

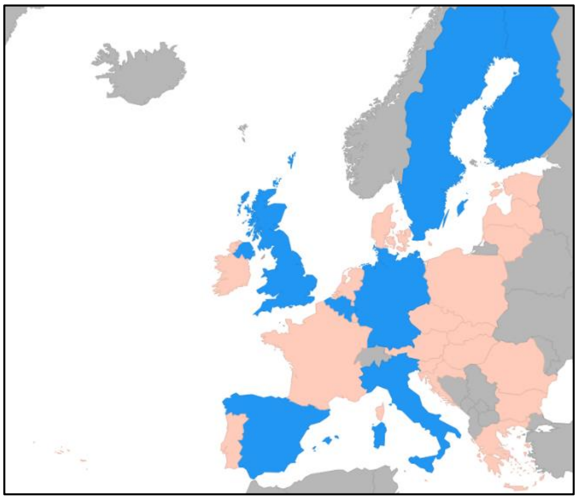


Fig 3.4. Countries with and without multivisceral transplants.

In blue, countries with multivisceral activity. In red, countries without it.

complicated, it is recommended for the surgery to take place in a more expert hospital, as it is La Paz University Hospital in Madrid (Spain), with 37 multivisceral procedures.

As the hospital locations in every country are shown in the map, it is possible to do a similar analysis nationally. As a case in point, if a patient from Bilbao (Spain), needs to be transplanted, the operation could be carried out in any of La Paz University Hospital in Madrid (Spain) or Vall d'Hebron University Hospital in Barcelona (Spain) because they are at a similar distance. Analogously to the previous case, La Paz University Hospital is a better option because of its experience. These proposed solutions are sketched in Fig 3.5.

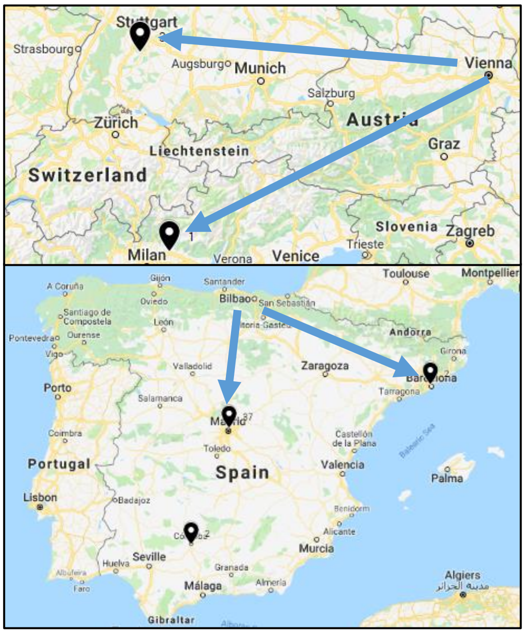


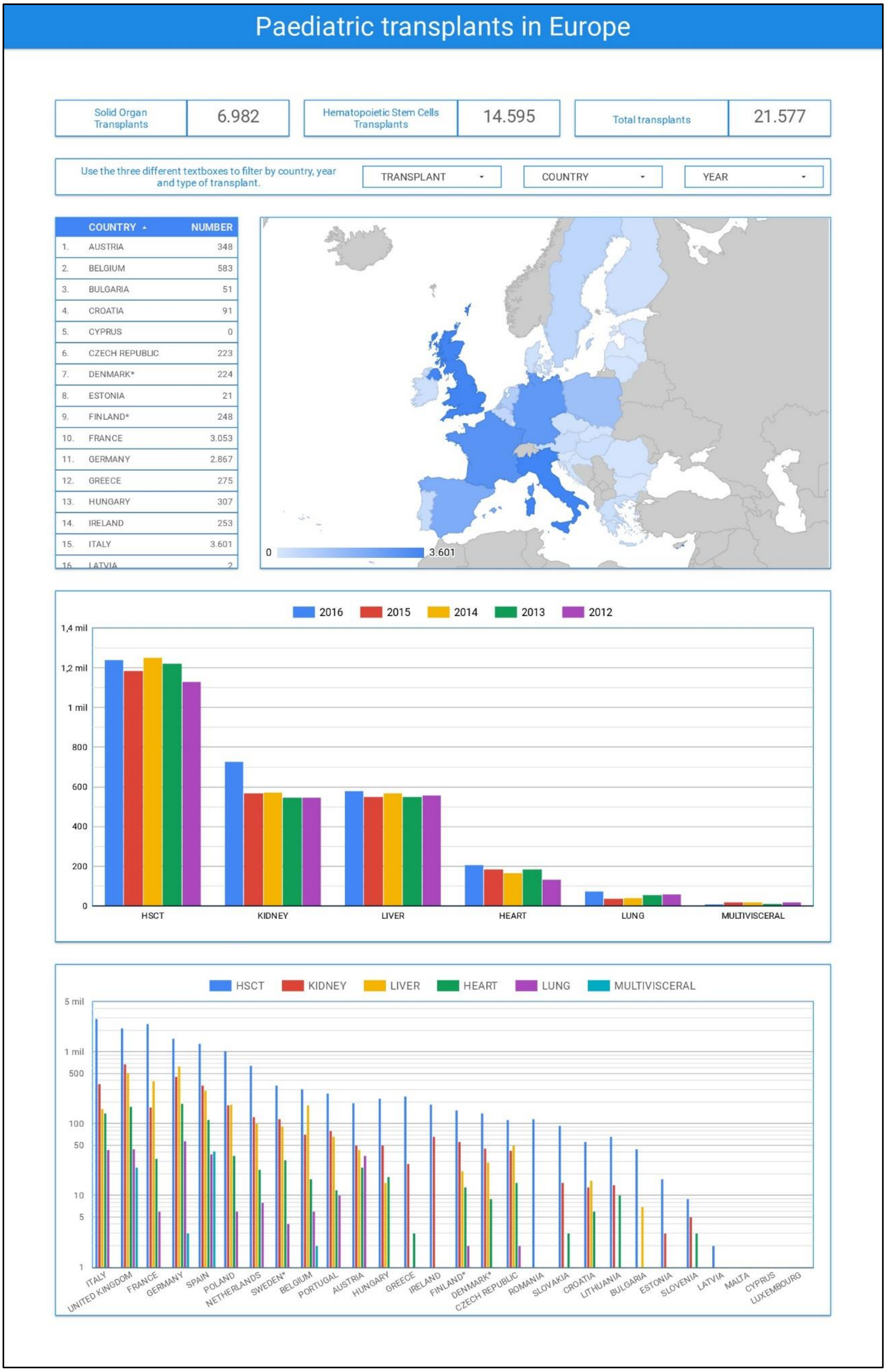
Fig 3.5. Possible transplant solutions internationally and nationally.

3.1.2. Paediatric transplants in the European Union reports

Three different reports were generated in Google Data Studio utilizing the data organization explained in 2.1.2.2. Together, they offer a complete analysis of all the information, as each of them focuses on different aspects.

3.1.2.1. Paediatric transplants per country

The first interactive report describes the number of paediatric transplants performed in each country. For that purpose, four different plots were built, plus three textboxes indicating the number of SOT, HSCT and total transplants. The first graph utilized was a bar chart per countries and subdivided by transplant type, which was useful to compare among countries, and at the same time to be able to see in which type of transplant a nation stands out. Then, there was another similar plot, but this time it was separated by transplant types and by year of transplant simultaneously. It resulted helpful to see the contrast between the different types, but also to check the differences per year. The next graph consisted in a European map in which a colourmap was used to distinguish the countries and zones with more activity. Moreover, it is possible to filter in every feature by type of transplant, country and year.



Paediatric transplants in Europe

Solid Organ Transplants	6.982	Hematopoietic Stem Cells Transplants	14.595	Total transplants	21.577

Use the three different textboxes to filter by country, year and type of transplant.

TRANSPLANT ▾ COUNTRY ▾ YEAR ▾

	COUNTRY ▲	NUMBER
1.	AUSTRIA	348
2.	BELGIUM	583
3.	BULGARIA	51
4.	CROATIA	91
5.	CYPRUS	0
6.	CZECH REPUBLIC	223
7.	DENMARK*	224
8.	ESTONIA	21
9.	FINLAND*	248
10.	FRANCE	3.053
11.	GERMANY	2.867
12.	GREECE	275
13.	HUNGARY	307
14.	IRELAND	253
15.	ITALY	3.601
16.	LATVIA	2

Fig 3.6. Paediatric transplants per country: First report.

The report in Fig 3.6 facilitated the achievement of the following conclusions[12]:

- 21,577 transplants were performed in the EU between 2012 and 2016.
- HSCT outnumbers SOT in more than double: 14,595 to 6,982.
- Italy is the country with the highest number of transplants, having performed a total of 3,601.
- If the filters per year are applied, it can be observed an increase in the number of transplants in 2016. It augmented an 11.5% when compared to the average of the previous years.
- Focusing on SOTs (excluding HSCTs in the filters), United Kingdom (UK) and Germany are the leaders with 1,420 and 1,340 interventions respectively.
- Spain is the most outstanding country in terms of multivisceral transplantation with 41 procedures.
- Even with some missing France data, Central Europe is the region where most transplants are performed (Italy, UK, Germany, France and Spain): 15,242.

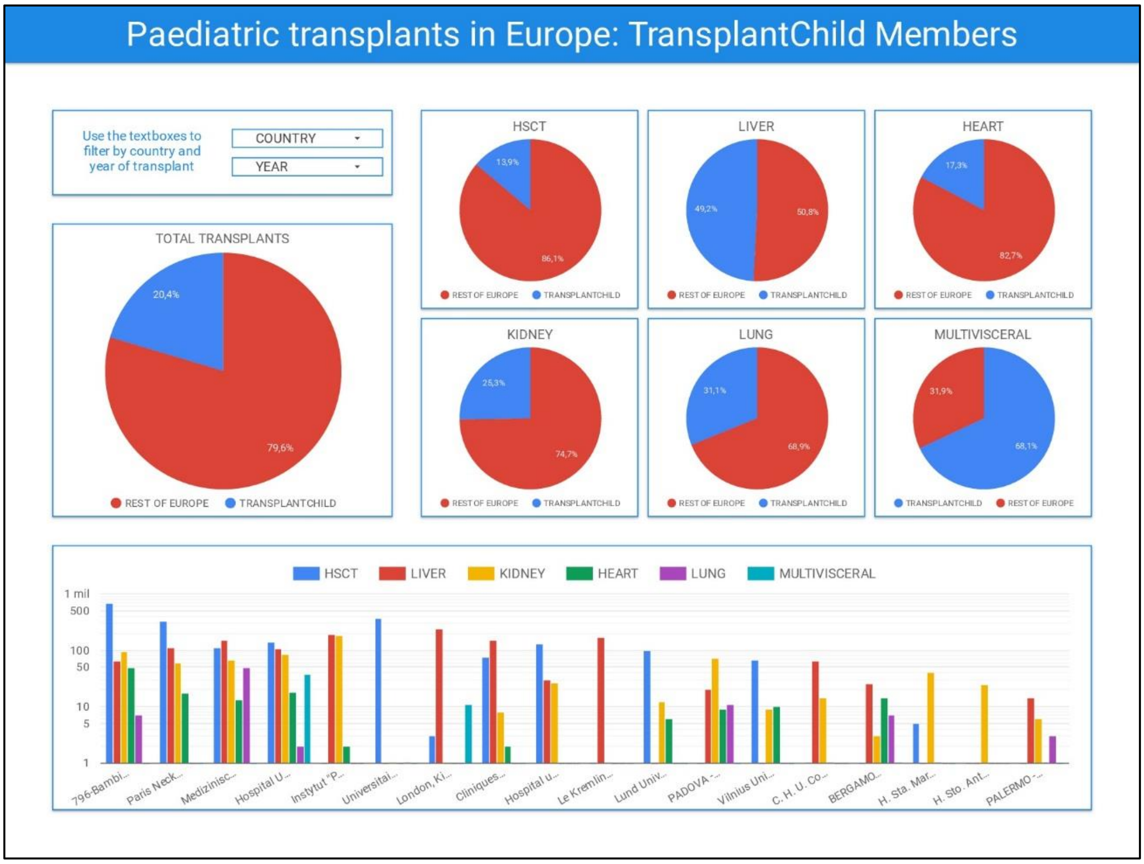
3.1.2.2. TransplantChild activity in the European Union

The second report had the objective of estimating the importance of TransplantChild in the EU in terms of activity. For that purpose, pie charts were used: one for every type of transplant, plus one for the sum of all. In every pie chart, it was represented the percentage of transplants performed by TransplantChild members versus the total transplants performed in the EU for that type.

Besides this group of graphs, the activity of all the centres belonging to the ERN was plotted in a separate bar chart. The reason behind this plot is the identification of the most important hospitals, and the transplant type in which each centre is specialized.

---

[12] Some of these conclusions can change when all the data is sent correctly, as the information from some countries was incomplete or missing.



Paediatric transplants in Europe: TransplantChild Members

Use the textboxes to filter by country and year of transplant

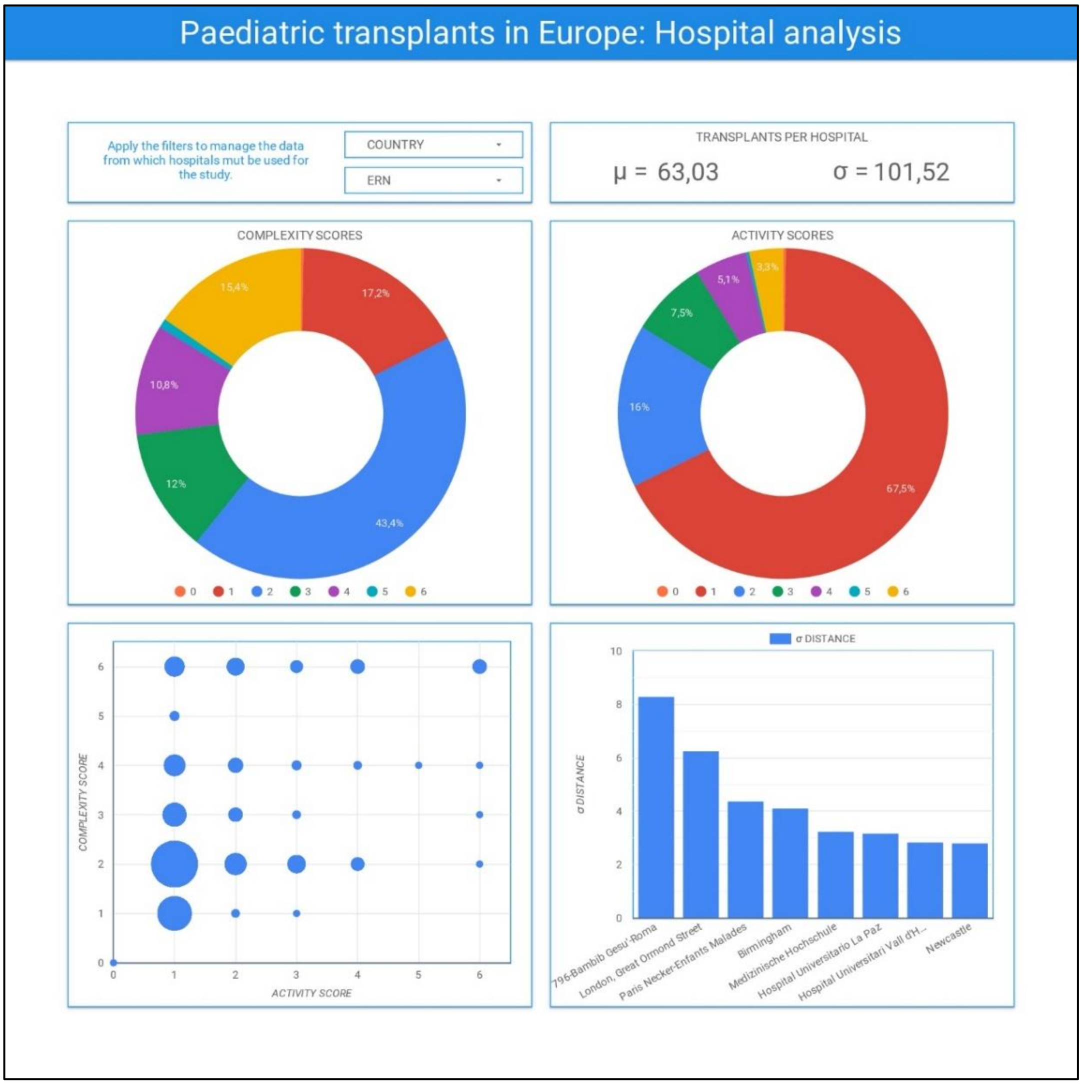
Fig 3.7. TransplantChild activity inside the EU: Second report.

Analysing the report in Fig 3.7, it is possible to distinguish:

- TransplantChild performs the 20.4% of all the paediatric transplants in the EU.
- It is remarkable that the network stands out in multivisceral transplants, carrying out a 68.1% (100% in Italy) of them.
- The hospitals with more PT activity in the network are:
    1. Bambino Gesù Hospital (Rome, Italy), with 908 transplants (216 SOTs).
    2. Necker-Enfants Malades Hospital (Paris, France), with 507 transplants (186 SOTs).
    3. Hannover Medical School (Hannover, Germany), with 392 transplants (280 SOTs).
    4. La Paz University Hospital (Madrid, Spain), with 387 transplants (248 SOTs, 37 of them multivisceral).

3.1.2.3. Hospital analysis

This report was based on building an analysis of all the hospitals with any transplantation activity, to obtain the best centres in Europe regarding activity and complexity. First, for that purpose, the transplants per hospital average μ and standard deviation σ are shown, to give a general idea of the distribution. Then, two pie charts were plotted: the first one contains the hospitals' complexity scores, and the second one the activity ones. After having checked that, in both plots, there were some centres with a score of 6, it was decided to create a scatter plot to see if there was any hospital with a maximum score in the two categories. As shown in Fig 3.8, there is a circle in the scatter plot indicating there are hospitals that fulfil the previous conditions. Hence, as a last step, a bar chart was generated in which only these hospitals with maximum score in both categories were included.



Paediatric transplants in Europe: Hospital analysis

Apply the filters to manage the data from which hospitals mut be used for the study.

COUNTRY

ERN

TRANSPLANTS PER HOSPITAL

μ = 63,03 σ = 101,52

Fig 3.8. Hospital analysis in the EU: Third report.

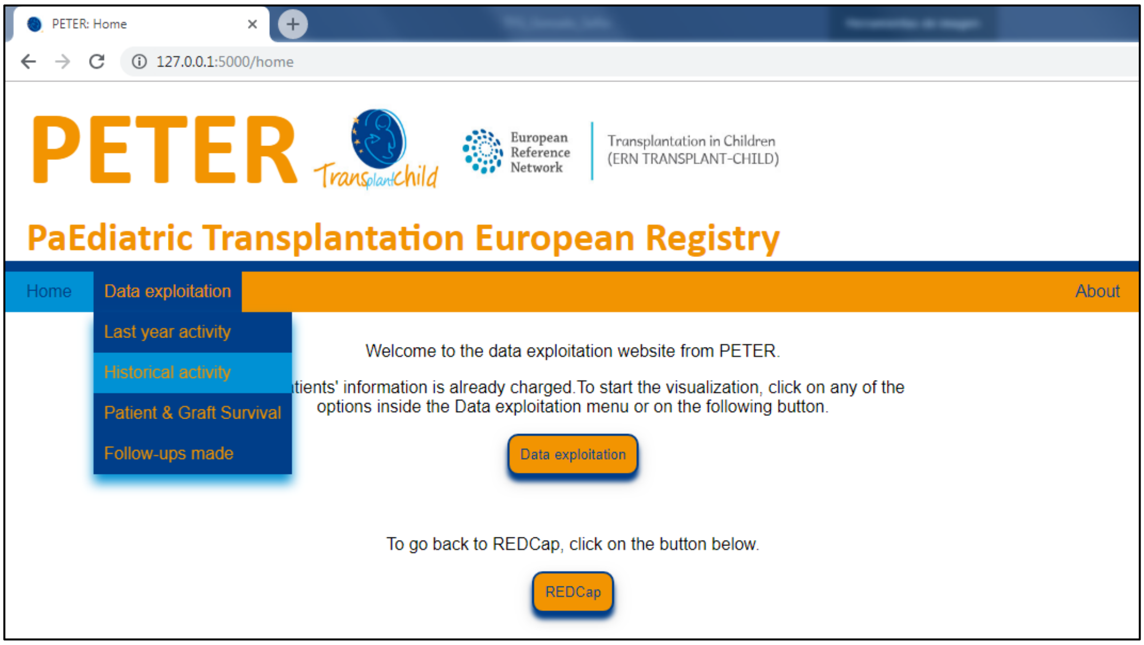
By analysing the report it is possible to obtain the following conclusions:

- On average, hospitals belonging to the EU performed 63 paediatric transplants from 2012 to 2016.
- Regarding complexity, the most common score is 2, meaning that 43.4% of the centres perform only HSCTs.
- There are 51 hospitals (15.4%) with HSCT and, at least, two types of SOT activity.
- With respect to activity, the majority of centres (67.5%) scored 1, which means they performed between 1 and 63 transplants.
- Only 11 hospitals obtained a score of 6, meaning they had an activity higher than 316 procedures.
- At the end, just 8 hospitals obtained the maximum score in both categories.
- From these expert centres, 4 are TransplantChild members: Bambino Gesù Hospital, Necker-Enfants Malades Hospital, Hannover Medical School and La Paz University Hospital.

3.2. PaEdiatric Transplantation European Registry

The results obtained for the registry consisted on the REDCap platform being linked to the developed website with a project bookmark. By applying the code developed in section 2.2.3, a multi-route application able to export the collected data and visualize them was constructed.

After accessing it through REDCap and introducing the necessary information, the user is taken to Home, where instructions are given to start using the website. If the REDCap button is clicked on, the REDCap project is returned again. Otherwise, to begin with data exploitation, the user must click on the Data exploitation button or in the any of the dropdown options in the Data exploitation menu, that is only shown when hovered over, as in Fig 3.9.



Welcome to the data exploitation website from PETER.

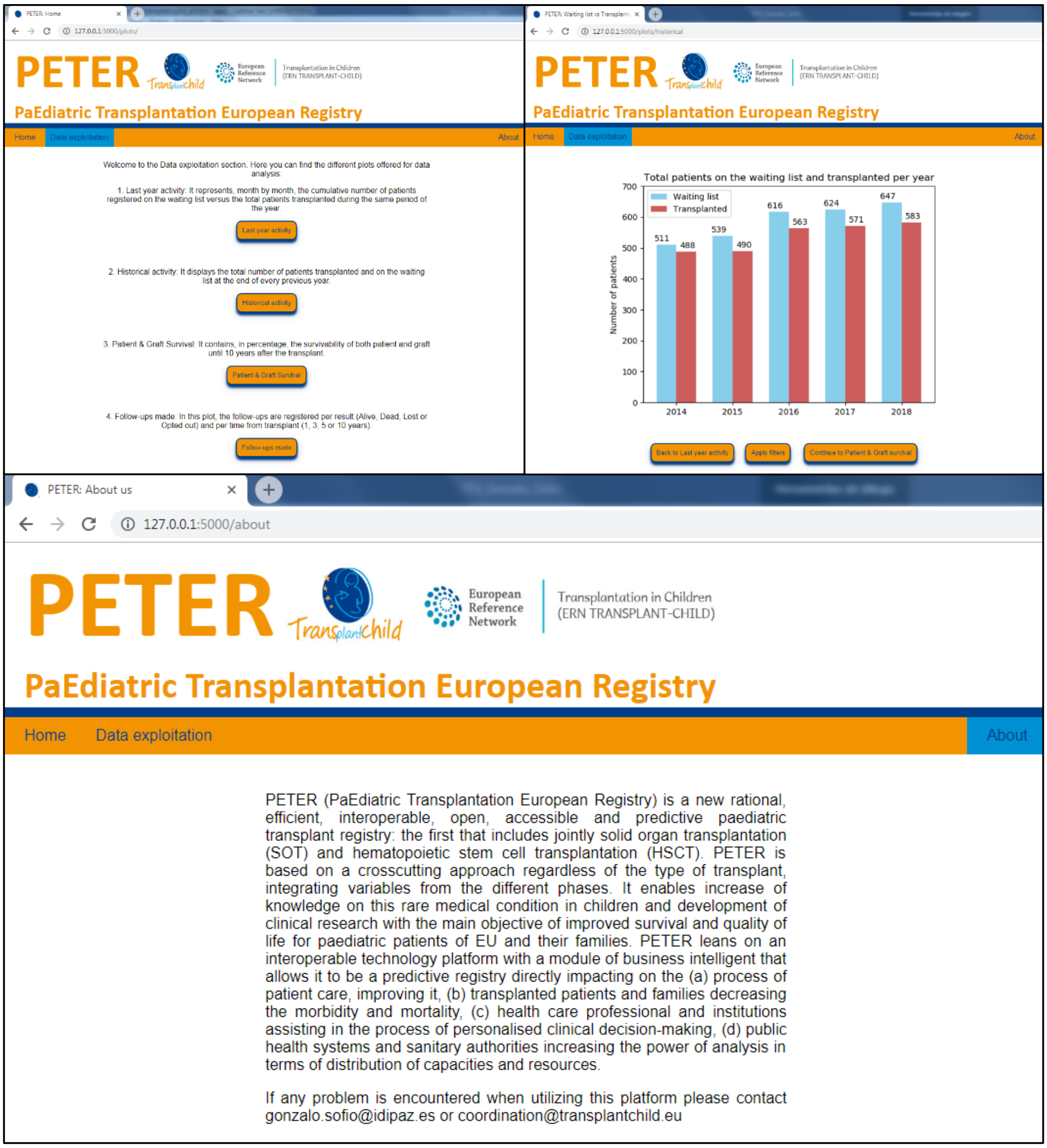
tients' information is already charged.To start the visualization, click on any of the options inside the Data exploitation menu or on the following button.

To go back to REDCap, click on the button below.

Fig 3.9. Home section with dropdown shown.

Once inside the */plots/* direction, there are individual descriptions of the four graphs introduced by now, with buttons linking to each of them. The first button corresponds to the cumulative number of patients both on the waiting list and transplanted during the last year. Secondly, the values obtained at the end of the previous year are compared with the ones of earlier years. Then patient and graft survival rate (in percentage are studied). To finish, there is a registration of the numbers and types of follow-ups made. When one of these graphs is selected, a redirection takes place to the indicated URL, and this time, in the body of the page, a plot image is displayed, together with the respective buttons containing the filters and the navigation between graphs.

The last section, About, outputs a project description and the contact details to help with any problem. It can be observed that the header is always kept the same, except for the active section marked in light blue. This direction is displayed in Fig 3.10, together with Data exploitation and Historical activity.



Welcome to the Data exploitation section. Here you can find the different plots offered for data analysis:

1. Last year activity: It represents, month by month, the cumulative number of patients registered on the waiting list versus the total patients transplanted during the same period of the year.

2. Historical activity: It displays the total number of patients transplanted and on the waiting list at the end of every previous year.

3. Patient & Graft Survival: It contains, in percentage, the survivability of both patient and graft until 10 years after the transplant.

4. Follow-ups made: In this plot, the follow-ups are registered per result (Alive, Dead, Lost or Opted out) and per time from transplant (1, 3, 5 or 10 years).

PETER (PaEdiatric Transplantation European Registry) is a new rational, efficient, interoperable, open, accessible and predictive paediatric transplant registry: the first that includes jointly solid organ transplantation (SOT) and hematopoietic stem cell transplantation (HSCT). PETER is based on a crosscutting approach regardless of the type of transplant, integrating variables from the different phases. It enables increase of knowledge on this rare medical condition in children and development of clinical research with the main objective of improved survival and quality of life for paediatric patients of EU and their families. PETER leans on an interoperable technology platform with a module of business intelligent that allows it to be a predictive registry directly impacting on the (a) process of patient care, improving it, (b) transplanted patients and families decreasing the morbidity and mortality, (c) health care professional and institutions assisting in the process of personalised clinical decision-making, (d) public health systems and sanitary authorities increasing the power of analysis in terms of distribution of capacities and resources.

If any problem is encountered when utilizing this platform please contact gonzalo.sofio@idipaz.es or coordination@transplantchild.eu

Fig 3.10. Website routes.

A) Data exploitation. B) Historical activity. C) About.

4. DISCUSSION

4.1. Budget, socioeconomic impact & regulation

Apart from the proper development of the work, there exist many other aspects that are critical for a study to be correctly carried out. It is necessary to obtain an estimation of the project-related costs: the budget. But this investment must be justified by explaining the possible impact that the undertaking can have on the society, and what improvements can be made. At last, the legal aspects and regulations must be considered to ensure that all the licenses and legislations are respected.

4.1.1. Budget

To obtain an estimation of the resources invested during the development of the thesis, they were divided in four categories: material, internet, software and personnel.

Regarding the material, the only necessary item was a computer. However, it was separated in their different components to get a more accurate evaluation. Then, each element was assigned a corresponding generic average price. Also, an internet connection was needed for the duration of the project (5 months). The standard cost per month was consulted, obtaining an estimation similar to the previous one with the material.

In this study, many online tools and software were utilized to develop the analyses and applications required. Even though many of them were free, some licenses were not. As a case in point, the Windows 10 Pro individual license is worth 259 €. Also, the Office 365 Business Premium has a cost of 10.50 € per month, and it was used for 5 months.

Different people collaborated in this project, so the number of working hours and their cost must be computed:

- The student, who worked for 580 hours: 330 hours of internships and 250 hours of thesis development. These hours were computed as 12.5 € each.
- The tutor in TransplantChild, who dedicated approximately one day per week to help in the thesis, so the estimation of the number of hours was calculated with the student hours divided by 5. They were valued in 25 € per hour.
- The tutor in UC3M, that was visited once every two weeks, dedicating approximately 2 hours in each tutorship. Apart from that, an estimation of 20 extra

hours were applied for document revision in the last week. This made a total of 40 hours at 25 € each.

- Lastly, when performing data collection, the work of a doctor or scientist of every institution was needed. Making an approximation of 2 working hours to group, structure their data in the correct way and send it, and considering that 24 countries provided information, that resulted in a total of 48 hours, with a cost of 25 € each.

TABLE 4.1. ESTIMATED PROJECT BUDGET.

Description	**Number**	**Individual cost**	**Subtotal cost**
Material	Units		
CPU	1	500 €	500 €
Screen	1	100 €	100 €
Keyboard	1	10 €	10 €
Mouse	1	10 €	10 €
Ethernet wire	1	10 €	10 €
Internet	Months		
ADSL connection	5	50 €	250 €
Software	Licenses		
Windows 10 Pro	1	259 €	259 €
Office 365 Business Premium	1	53 €	53 €
Google My Maps	1	0 €	0 €
Data Studio	1	0 €	0 €
REDCap	1	0 €	0 €
Python	1	0 €	0 €
Flask	1	0 €	0 €
PythonAnywhere	1	0 €	0 €
Personnel	Hours		
Tutor 1	116	25 €	2.900 €
Tutor 2	40	25 €	1.000 €
Student	580	13 €	7.250 €
Data collection	48	25 €	1.200 €
Total			13.542 €

By adding all of these costs as represented in Table 4.1, an estimated project investment of 13.542 € was obtained. This is a relatively low amount, as the majority of tools used for the study were free. In fact, the main expenses were the personnel salaries, resulting in the 91% of the total cost.

4.1.2. Socioeconomic impact

Transplanted children's well-being gets affected by transplantation in all of their physical, mental and social aspect. In the current European society, family has been consolidated as the most important unit, so when a transplant is performed, the condition does not affect only the patient, but the whole family, in different ways.

First, receiving a SOT increases stress on patients and family members. Psychological outcomes can be measured to find possible interventions and improvements. In several studies, it was found that parents of patients usually suffer from elevated stress and Post Traumatic Stress Disorder (PTSD). Family functioning was registered, getting to the conclusion of it being associated with factors like medication adherence or hospitalizations. These symptoms are present also after transplantation, resulting in an important target to be researched [46].

Moreover, in a context of PT, conjugal resilience[13] may be significantly damaged. The transplantation treatments, in both SOT and HSCT, interfere with everyday life, often making the family feel impotent towards the chronic condition provoked by the transplant procedure. This feeling debilitates couple cohesion and the previously mentioned concept of conjugal resilience [17]. Also, "couples set aside certain elements that would usually define their relationship so as to dedicate it exclusively to the care of the child, resulting in decreased intimacy and sexual desire, lessening of expectations that they would normally have for each other about affection, and so on" [47], even getting to lose the concept of actual 'couple' to substitute it by 'team' or 'fighters'.

---

[13] Conjugal resilience is defined "as a characteristic of the couple in which conjugal satisfaction, partners' affinity and mutual respect, and the trust they have in their relationship, are preserved or strengthened through adversity" [47].

Besides this, the patient's situation can have an impact on other familiar aspects, like psychologically affecting siblings too. They may suffer from various problems like lacking information or feeling excluded [17].

With this project, it is aimed to improve all the mentioned social aspects of transplantation by helping its research across the continent. If the procedure is perfected and its post-transplant complications are minimized, it will be possible to significantly decrease the prevalence of these social impairments for both the patient and the family. Moreover, by improving the technique, patient survival will be improved, which is, in the end, the final goal.

The mentioned improvements will be achieved by obtaining reference values from expert hospitals, that facilitate the benchmarking among transplantation centres, so that it results easier to identify possible weaknesses and act in consequence [48].

Regarding the economic aspect, in 2013 the Annual Growth Survey acknowledged the role of healthcare in tackling poverty. The economic beneficial effect of improving health systems was recognised, as this could help facilitating their cost-effectiveness [17]. This project can help in several manners to guarantee the sustainability of the EU healthcare system.

On the one hand, by helping research, it is possible to increase the quality of transplants and eliminate possible chronic complications and adverse effects. This would help saving resources in number of hospitalizations, surgeries (retransplantations) and medicines needed. Also, increasing patient survival can provoke a positive economic impact: if transplanted children live enough to grow up and have a job, an income is generated. In fact, according to a study carried out in France in 2014, 53% of the participant patients had a job at that time [49]. Moreover, the less complications they suffer, the less unemployment and disability [50]. Hence, success in transplant procedures can directly have a positive economic impact in society.

On the other hand, implementing PETER would make a unique registry for all the hospitals in TransplantChild. If this project was not carried out, every hospital, institution or country would need to build a specific registry, investing unnecessary resources in elaborating different platforms and formats. However, in this way, it is possible to unify many hospitals and make everything interoperable in a unique registry, resulting in a much lower investment Also, this can be applicable for many more healthcare centres if they join the network or the registry in the future, contributing to the previously mentioned benchmarking.

4.1.3. Regulation

4.1.3.1. Licenses

It was necessary to pay attention to the software used and ensure all the conditions were satisfied in order for its licensed and fair use.

- Google My Maps: It is allowed to use printed images for professional documents like research papers and reports. It is required to do a correct attribution to Google and their data providers [51]. In the images used in this thesis, both the Google My Maps logo and the attribution information are included.
- Google Data Studio: No license is needed. The only obligations are: being responsible of the End Users activity, ensuring an Acceptable Use and following the General Data Protection Regulation (GDPR) [52].
- REDCap: Even though the license is free, it must be approved by Vanderbilt. It is granted under the following conditions [53]:
    a. The software must neither shared nor distributed for any purpose.
    b. The software is installed on servers owned by the licensee or in the cloud only if the hosting company does not have access to REDCap, and it is protected at every time.
    c. It is allowed the limited access of a collaborator to introduce data, but not for any other use.
    d. With the release of new REDCap versions, older ones may become obsolete, receiving no support or being unavailable for its use.
- Python: The Python Software Foundation (PSF) establishes a free license based on accepting their terms and conditions. “PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyse, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 3.7.2 alone or in any derivative version” [54].

- Flask: This microframework has a three clause BSD (Berkeley Software Distribution) License. This license consists on: “Redistribution and use in source and binary forms of the software as well as documentation, with or without modification, are permitted provided that the following conditions are met” [55]:
    a. The copyright notice, the license conditions and the disclaimer must be included.
    b. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
    c. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
- NumPy & pandas: They both have a BSD license, allowing its reuse under some conditions, very similarly to the Flask license [40].
- Matplotlib: It is licensed based on the PSF license explained for Python [42].

4.1.3.2. Bioethics

For a transplant to be performed, a compatible donor must be found. This should always be a non-discriminative process. Therefore, personal relations must be kept out. To enhance this and avoid any conflicts to appear, TransplantChild members must commit to the rules stated [17]:

- Do not give any priority to patients belonging to the network or the registry.
- Always follow the regulatory guidelines, giving no priority under any condition like, for example, subornation.
- Keep data protected and properly encrypted at every step of the communication between ERN members.
- Members of the network should always communicate if a patient is any type of relative.
- Avoid every type of bias: religious, cultural, racial, etc. None of them are permitted.

For some types of transplants, the organ is always received from a deceased donor, as that organ is necessary for life. This is the case of heart, lung and multivisceral transplants. However, in kidney and liver transplants, the donor may be living and, in fact, in allogeneic HSCTs, the donor is always living (in autologous too, but it is the same person).

In these last cases, the patient receives the organ from a non-profit and anonymous donor, except when the donor is a relative, which is usually recommended to facilitate compatibility. This experience, apart from saving and changing patients' lives like every successful transplant, is also emotionally rewarding for the donors. A satisfaction feeling is usually provoked on them, because of knowing that another person's life was saved utilizing the donated organ or tissue.

4.1.3.3. Data protection

All data protection policies were unified by the EU in a single law called GDPR. In it, new guidelines and rules are implemented regarding data globalisation, social media and technological advances [56]. It is expected to promote data security and their correct processing in the EU. According to this new regulation, the PETER project, as it contains patients' data, must follow these aspects [17]:

- Ensure patients have the rights of data access, portability and transparency, as they are the data owners.
- Adapt to the Regulation to properly carry out the registry, so that research is not impeded.
- Enhance cooperation by establishing an equivalent level of data protection across EU, promoting global research.
- Ensure the compliance of ethical aspects, by being transparent and involving patients in decision-making related to their personal data.

4.2. Limitations

Despite the usefulness of the projects, several constraints were encountered during its development, which limited the study final results. For a better understanding and organization, the limitations regarding the analysis of PT activity were stated first:

- The collected information presented a lot of imperfections, since just 5 from the 27 countries sent the information as indicated. Because of this, when using the data from the transplant newsletter for verification, the comparison showed many inequalities that should not exist.
- Google My Maps does not have a system for automatically updating data, requiring a manual modification every time new information needs to be introduced. Because of this, it resulted impossible to store all the data in a unique file that could automatically refresh the rest of applications.
- Google My Maps did not facilitate the elaboration of a quantitative statistical analysis, restricting it to a qualitative examination of the transplantation centres geographical distribution. It also did not offer the possibility of showing more than one layer at the same time without them overlaying each other.
- With respect to the platform development, some problems were encountered during the work, and constraints were present too, making the project more difficult because of the adding of extra conditions to be fulfilled:
- The restriction of using REDCap as data collection platform conditioned the software exploration: first, it was not valid as data exploitation platform and, second, the platform developed needed to be integrated with it.
- The fact that it was not possible to obtain a TransplantChild REDCap. As the REDCap server used belongs to IdiPAZ, some of the necessary privileges were lacking.
- The project budget also conditioned software selection: LabKey could have been a suitable solution if it had not been because of its price.
- Data collection had not been performed yet, which prevented the full website implementation because of the lack of real information to be exploited. That is why this was done with randomly generated data.
- The system for patient pseudonymization had not been put into effect yet. This is a necessary feature in all registries so that patients' information is protected during its collection and exploitation.

4.3. Future

With the aim of solving the study's problems and overriding the previous limitations, many solutions were thought to be implemented in the project near future. They were subdivided according to the two parts of the project.

4.3.1. Analysis of paediatric transplantation activity in the European Union

To improve the quality of the obtained data, the comparison between the collected and the newsletter's information was sent to all the collaborating countries, with a brief explanation of the meaning and the possible reasons behind the inaccuracies. Hence, it is expected that the countries give feedback, completing their numbers or explaining why they are different.

Then, the map and the reports will be updated with the new information, so that either the obtained conclusions are confirmed, or new ones are achieved. When this is done, they will be posted on the TransplantChild website, *https://www.transplantchild.eu/en/*. Therefore, all members of the network (and all users who access this page) will be able to check and explore the represented data.

4.3.2. PaEdiatric Transplantation European Registry

The next step in the development of PETER is performing data collection. However, the current project structure does not facilitate this process workflow: it needs to be simplified to promote an easy registration, reducing their variables to the most important ones.

When real data are available, it is necessary to deploy the platform to the cloud so the exploitation can begin. The idea is to incorporate the created application inside the TransplantChild website and utilize its server to store the information coming from REDCap, instead of loading it in every session.

On the other hand, if TransplantChild receives their own REDCap server, the project will be moved to it. If this happens, the external modules would be implemented using the administrator account. Otherwise, if REDCap is not provided to the ERN, the most suitable option for the future would be to fully develop the platform from scratch, integrating data collection inside the already developed website.

Lastly, it is necessary to implement a system for patient pseudonymization[14]. This will be put into effect using EUPID.

4.3.2.1. EUPID

EUPID is a patient ID management system thought to "facilitate secondary use of datasets in Biomedical Research and Healthcare" [57], providing an implementable system for computing, including cloud, environments [58].This tool, whose scheme is indicated in Fig 4.1, presents the following benefits to registries that utilize it [59]:

- GDPR compliance is guaranteed.
- Apply pseudonymization to protect data.
- Having a private designed, only provided to Trusted Third Parties (TTP).
- Minimize personal information processing: these data must be sent in an encrypted way.
- Application of Privacy Preserving Record Linkage (PPRL) to allow "a (future) merge of data from different sources e.g. different RD Registries" [59].

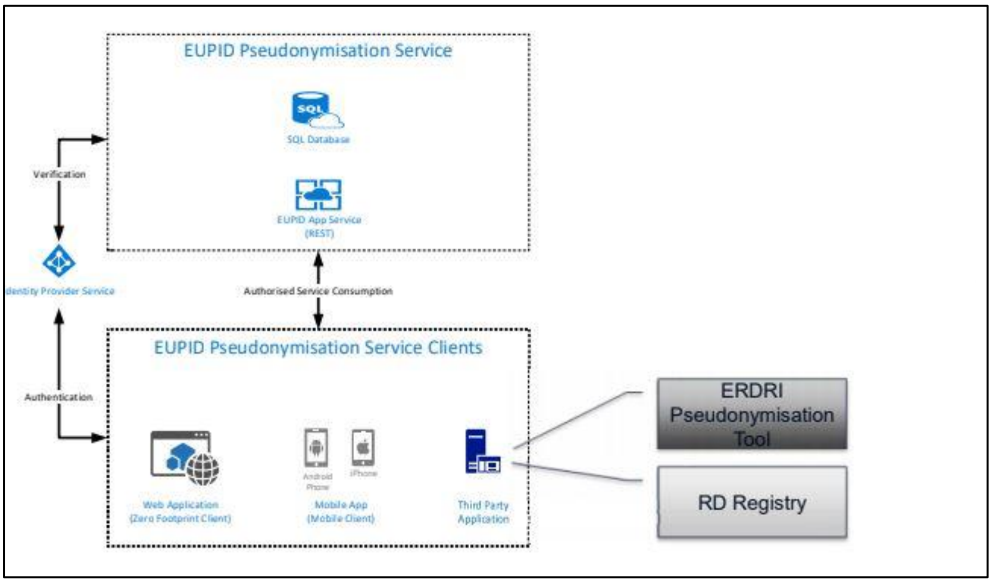


Fig 4.1. Patient Identity Management: EUPID working scheme [59].

---

[14] Pseudonymization "means the processing of personal data in such a manner that the personal data can no longer be attributed to a specific data subject without the use of additional information, provided that such additional information is kept separately and is subject to technical and organisational measures".

4.4. Conclusions

The different tools and applications generated in this project provide a specific analysis of the PT activity situation in the European Union. With them, it is possible to identify transplantation issues and fix them or, at least, propose feasible solutions. When PETER gets put into practice, data collected in this registry will be unified with the activity information, so that the complete analysis is integrated.

Considering the state of the art, this project settles a new solution to keep improving PT results through research and data analysis. Similar studies have been carried out for adult transplantation, but with the development of this thesis and its future improvements, it is intended to amend the lack of PT registration.

Therefore, by promoting the elaboration of a European benchmarking for PT activity, an indirect increase in the procedure’s success rate will be produced, and the technique of transplantation will be perfected in the long term. This will result in a social improvement for patients and families, as well as a positive economic impact related to surviving patients developing a labour life. In the end, this objective of increasing patients’ quality of life and survivability is the final goal of transplantation and medicine.

5. REFERENCES

[1] J. M. Eyler, “Smallpox in history: The birth, death, and impact of a dread disease,” *J. Lab. Clin. Med.*, vol. 142, no. 4, pp. 216–220, 2003.

[2] J. G. Breman and I. Arita, “The confirmation and maintenance of smallpox eradication,” *World Heal. Organ.*, vol. 156, pp. 1–22, 1980.

[3] T. Miloh *et al.*, “Immunosuppression in pediatric liver transplant recipients: Unique aspects,” *Liver Transplant.*, vol. 23, no. 2, pp. 244–256, 2017.

[4] C. L. Bryce *et al.*, “Sociodemographic Differences in Early Access to Liver Transplantation Services,” *Am J Transpl.*, vol. 9, no. 9, pp. 2092–2101, 2009.

[5] D. H. Adams, A. Sanchez-Fueyo, and D. Samuel, “From immunosuppression to tolerance,” *J. Hepatol.*, vol. 62, no. S1, pp. S170–S185, 2015.

[6] E. Hatzimichael and M. Tuthill, “Stem Cells and Cloning: Advances and Applications Dovepress Hematopoietic stem cell transplantation,” *Stem Cells Cloning Adv. Appl.*, pp. 3–105, 2010.

[7] J. R. Passweg *et al.*, “Hematopoietic stem cell transplantation in Europe 2014: More than 40 000 transplants annually,” *Bone Marrow Transplant.*, vol. 51, no. 6, pp. 786–792, 2016.

[8] J. Burman, A. Tolf, H. Hägglund, and H. Askmark, “Autologous haematopoietic stem cell transplantation for neurological diseases.,” *J. Neurol. Neurosurg. Psychiatry*, vol. 89, no. 2, pp. 147–155, Feb. 2018.

[9] S. Bharadwaj *et al.*, “Current status of intestinal and multivisceral transplantation.”

[10] D. Van Esso *et al.*, “Paediatric primary care in Europe: Variation between countries,” *Arch. Dis. Child.*, vol. 95, no. 10, pp. 791–795, 2010.

[11] J. C. Magee, J. C. Bucuvalas, D. G. Farmer, W. E. Harmon, T. E. Hulbert-Shearon, and E. N. Mendeloff, “Pediatric transplantation,” *Am. J. Transplant.*, vol. 4, pp. 54–71, Apr. 2004.

[12] B. López Álvarez, C. León Canseco, and J. M. Torres Canizales, “Implementation Tools for Data Integration and Knowledge Sharing in the European Reference Network: Transplantchild.,” Universidad Carlos III de Madrid.

[13] M. Zaletel, M. Kralj, M. Magajne, and P. Doupi, *Methodological Guidelines and*

Recommendations for Efficient and Rationale Governance of Patient Registries, vol. 25, no. suppl_3. 2015.

[14] EU, *European Reference Networks*. 2017.

[15] "TransplantChild Background." [Online]. Available: https://www.transplantchild.eu/en/about-transplantchild/transplantchild-background. [Accessed: 08-Jan-2019].

[16] "Members of the Network." [Online]. Available: https://www.transplantchild.eu/en/about-transplantchild/members-of-the-network. [Accessed: 08-Jan-2019].

[17] E. Commission, "3rd Health Programme Call : HP-PJ-06-2016 (Rare diseases - support for New Registries) Topic : PJ-06-2016 Type of action : HP-PJ Proposal number : 777333 Proposal acronym : PETER Deadline Id : HP-PJ-06-2016 Table of contents," 2016.

[18] EUCERD, "Eucerd Core Recommendations on Rare Disease Patient Registration and Data Collectionto the European Commission, Member Statesand All Stakeholders," *EUCERD Core Eur. Union Comm. Experrts Rare Dis.*, no. June, pp. 1–6, 2013.

[19] N. Lamba, D. Holsgrove, and M. L. Broekman, "The history of head transplantation: a review," *Acta Neurochir. (Wien)*.

[20] "Paediatric Transplantation." [Online]. Available: https://www.transplantchild.eu/en/about-transplantchild/about-pediatric-transplant. [Accessed: 05-Feb-2019].

[21] B. Domínguez-Gil and R. Matesanz, "Newsletter Transplant: International figures on donation and transplantation 2017," 2018.

[22] "My Maps Help." [Online]. Available: https://support.google.com/mymaps/?hl=en#topic=3188329. [Accessed: 05-Feb-2019].

[23] "Data Studio Product Overview." [Online]. Available: https://datastudio.google.com/overview. [Accessed: 21-Jan-2019].

[24] P. A. Harris, R. Taylor, R. Thielke, J. Payne, N. Gonzalez, and J. G. Conde,

"Research electronic data capture (REDCap)-A metadata-driven methodology and workflow process for providing translational research informatics support," *J. Biomed. Inform.*, vol. 42, pp. 377–381, 2008.

[25] V. U. M. Center, "External Modules: Module Review Guidelines."

[26] P. Chase, "GitHub - ctsit_redcap_user_profile_ REDCap external module that provides user profiles management," 2017. [Online]. Available: https://github.com/ctsit/redcap_user_profile/releases. [Accessed: 28-Jan-2019].

[27] L. Stevens, "GitHub - lsgs_redcap-autoschedule_ REDCap External Module for automated generation of record event schedules," 2018. [Online]. Available: https://github.com/lsgs/redcap-autoschedule. [Accessed: 28-Jan-2019].

[28] R. Taylor, "GitHub - vanderbilt-redcap_email-alerts-module_ Module to send customized emails alerts after a form_survey is completed to one or several recipients," 2017. [Online]. Available: https://github.com/vanderbilt-redcap/email-alerts-module/releases?after=REL_ORI07LP_20171030132629. [Accessed: 28-Jan-2019].

[29] M. Cormar, "GitHub - ctsit_redcap_css_injector_ Allow REDCap administrators to inject CSS into surveys and data entry forms," 2018. [Online]. Available: https://github.com/ctsit/redcap_css_injector. [Accessed: 28-Jan-2019].

[30] P. Chase, "GitHub - ctsit_form_render_skip_logic_ This REDCap module hides and shows instruments based on the value of a single field on a single form - i," 2017. [Online]. Available: https://github.com/ctsit/form_render_skip_logic. [Accessed: 28-Jan-2019].

[31] R. Taylor, "GitHub - vanderbilt-redcap_auto-dags-module_ REDCap External Module that automatically creates, renames, and assigns records to DAGs based on a specified field," 2017. [Online]. Available: https://github.com/vanderbilt-redcap/auto-dags-module. [Accessed: 28-Jan-2019].

[32] "Chronic disease management system | Marand." [Online]. Available: http://www.marand.com/pathfinder/. [Accessed: 14-Jan-2019].

[33] "Innovation, Experience, Agile, Marand | Marand." [Online]. Available: http://www.marand.com/en/why/#overview. [Accessed: 14-Jan-2019].

[34] E. K. Nelson *et al.*, "LabKey Server: An open source platform for scientific data

integration, analysis and collaboration," *BMC Bioinformatics,* vol. 12, 2011.

[35] "Explora JMP _ JMP." [Online]. Available: https://www.jmp.com/es_es/see-jmp.html. [Accessed: 09-Jan-2019].

[36] "Getting Started — transmartApp 16.2 documentation." [Online]. Available: https://transmart-app.readthedocs.io/en/latest/getting_started.html. [Accessed: 09-Jan-2019].

[37] "Installing tranSMART - tranSMART Project wiki - Confluence." [Online]. Available: https://wiki.transmartfoundation.org/display/transmartwiki/Installing+tranSMART. [Accessed: 09-Jan-2019].

[38] G. Van Rossum, "What is Python? Executive Summary | Python.org." [Online]. Available: https://www.python.org/doc/essays/blurb/. [Accessed: 21-Jan-2019].

[39] Python Software Foundation, "PyPI - the Python Package Index : Python Package Index," *Python*, 2018. [Online]. Available: https://pypi.python.org/pypi.

[40] "NumPy — NumPy." [Online]. Available: http://www.numpy.org/index.html. [Accessed: 28-Jan-2019].

[41] "The pandas project — pandas: Python Data Analysis Library." [Online]. Available: http://pandas.pydata.org/about.html. [Accessed: 28-Jan-2019].

[42] "Matplotlib: Python plotting — Matplotlib 3.0.2 documentation." [Online]. Available: https://matplotlib.org/. [Accessed: 28-Jan-2019].

[43] "Project Jupyter | About Us." [Online]. Available: https://jupyter.org/about. [Accessed: 22-Jan-2019].

[44] F. Perez, B. E. Granger, C. P. San, and L. Obispo, "Project Jupyter: Computational Narratives as the Engine of Collaborative Data Science," 2015.

[45] M. Grinberg, *Flask Web Development,* 1st ed. O'Reilly Media, Inc., 2014.

[46] M. K. Cousino, K. E. Rea, K. R. Schumacher, J. C. Magee, and E. M. Fredericks, "A systematic review of parent and family functioning in pediatric solid organ transplant populations," *Pediatr. Transplant.*, vol. 21, no. 3, p. e12900, May 2017.

[47] J. Martin, K. Péloquin, M.-F. Vachon, M. Duval, and S. Sultan, "Systemic aspects of conjugal resilience in couples with a child facing cancer and marrow

transplantation,” *Int. J. Qual. Stud. Health Well-being*, vol. 11, no. 1, p. 32423, Jan. 2016.

[48] X. Muller *et al.*, “Defining Benchmarks in Liver Transplantation,” *Ann. Surg.*, vol. 267, no. 3, pp. 419–425, Mar. 2018.

[49] H. Mellerio *et al.*, “Adult social and professional outcomes of pediatric renal transplant recipients,” *Transplantation*, vol. 97, no. 2, pp. 196–205, 2014.

[50] S. Rocha *et al.*, “Impact of pediatric kidney transplantation on long-term professional and social outcomes,” *Transplant. Proc.*, vol. 43, no. 1, pp. 120–124, 2011.

[51] “Permissions – Google.” [Online]. Available: https://www.google.com/permissions/geoguidelines/. [Accessed: 23-Jan-2019].

[52] “Terms of Service - Data Studio Help.” [Online]. Available: https://support.google.com/datastudio/answer/7019158?hl=en. [Accessed: 23-Jan-2019].

[53] “REDCap License Terms – REDCap.” [Online]. Available: https://projectredcap.org/partners/termsofuse/. [Accessed: 24-Jan-2019].

[54] “History and License — Python 3.7.2 documentation.” [Online]. Available: https://docs.python.org/3/license.html. [Accessed: 24-Jan-2019].

[55] “License — Flask 1.0.2 documentation.” [Online]. Available: http://flask.pocoo.org/docs/1.0/license/. [Accessed: 24-Jan-2019].

[56] “General Data Protection Regulation (GDPR): The paradigm shift in privacy (August, 2018),” 2018.

[57] “EUPID Patient Management.” [Online]. Available: https://eupid.eu/#/home. [Accessed: 24-Jan-2019].

[58] M. Nitzlnader and G. Schreier, “Patient Identity Management for Secondary Use of Biomedical Research Data in a Distributed Computing Environment,” 2014.

[59] M. Nitzlnader, “European Patient Identity Services (EUPID) ERDRI Pseudonymisation Tool,” 2018.

[60] M. St. Pierre and B. LaPlante, “Issues in Crosswalking Content Metadata Standards,” 1999.

[61] L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb, *Social Coding in GitHub: Transparency and Collaboration in an Open Software Repository Human Factors; Design*. 2012.

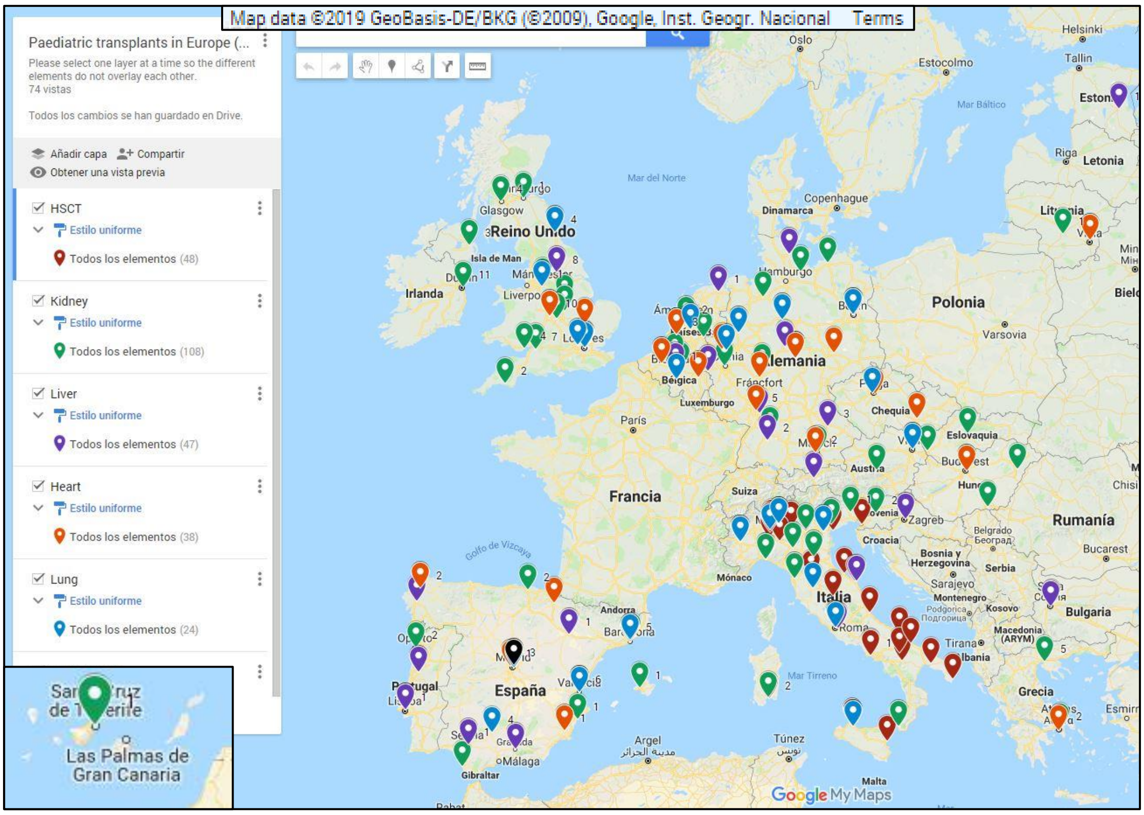
[62] “Versions and constraints - Composer,” *getcomposer.org*.

[63] D. Teodoro, E. Sundvall, M. J. Junior, P. Ruch, and S. M. Freire, “ORBDA: An openEHR benchmark dataset for performance assessment of electronic health record servers,” 2018.

[64] T. Berners-Lee, R. Fielding, and H. Frystyk, “Hypertext Transfer Protocol -- HTTP/1.0,” 1996.

[65] D. Niederwieser *et al.*, “Hematopoietic Stem Cell Transplantation Activity Worldwide in 2012 and a SWOT Analysis of the Worldwide Network for Blood and Marrow Transplantation Group (WBMT) including the global survey.”

ANNEX A: ACTIVITY MAPS PER AGE



Paediatric transplants in Europe (...

Please select one layer at a time so the different elements do not overlay each other.

74 vistas

Todos los cambios se han guardado en Drive.

☑ HSCT
Estilo uniforme
Todos los elementos (48)

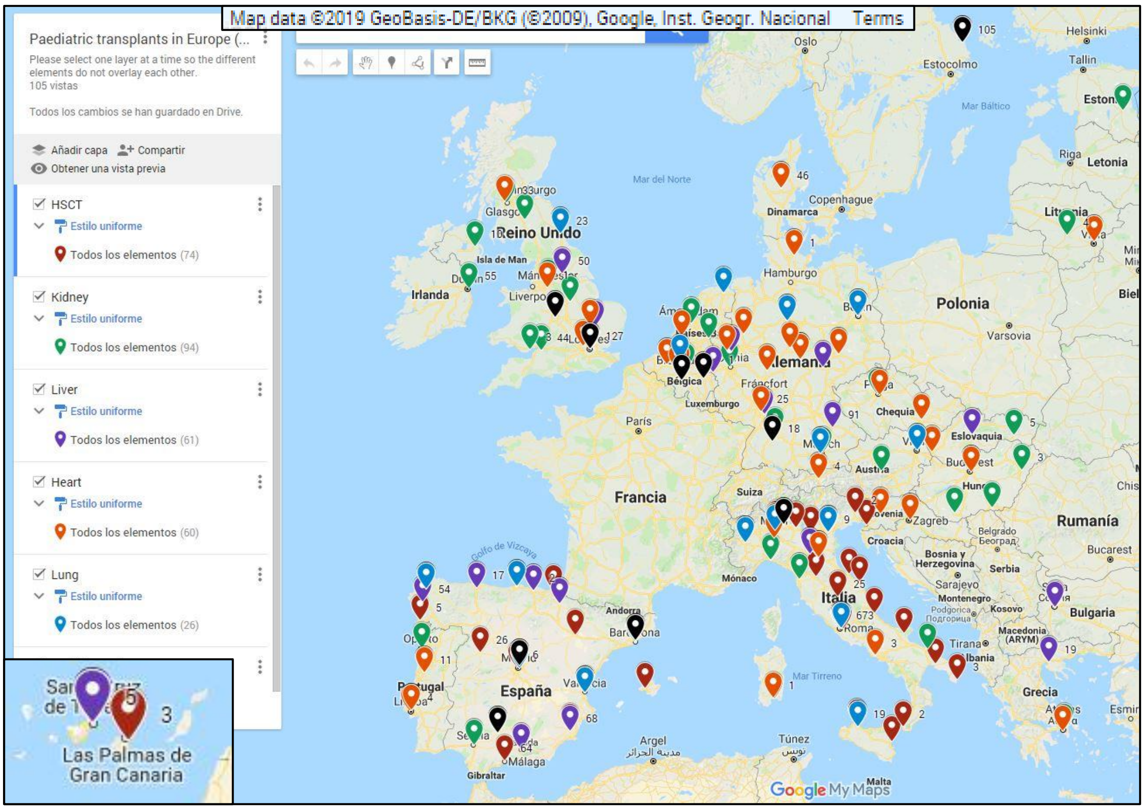
☑ Kidney
Estilo uniforme
Todos los elementos (108)

☑ Liver
Estilo uniforme
Todos los elementos (47)

☑ Heart
Estilo uniforme
Todos los elementos (38)

☑ Lung
Estilo uniforme
Todos los elementos (24)

0-16 years old map.



Paediatric transplants in Europe (...

Please select one layer at a time so the different elements do not overlay each other.

105 vistas

Todos los cambios se han guardado en Drive.

☑ HSCT
Estilo uniforme
Todos los elementos (74)

☑ Kidney
Estilo uniforme
Todos los elementos (94)

☑ Liver
Estilo uniforme
Todos los elementos (61)

☑ Heart
Estilo uniforme
Todos los elementos (60)

☑ Lung
Estilo uniforme
Todos los elementos (26)

17-18 years old map.

ANNEX B: WEBSITE FULL CODE

main.py:

```
import graphs
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
from flask import Flask
from flask import render_template
from flask import send_file
from flask import request
from flask import session
from requests import post
from config import DevelopmentConfig
from matplotlib.collections import EventCollection
from io import BytesIO

app = Flask(__name__)
app.config.from_object(DevelopmentConfig)

@app.route("/redcap", methods = ['GET', 'POST'])
def redcap():

    authkey = ''
    if request.method == 'POST':
        authkey = request.form['authkey']
        r = post('http://idipaz.org/api/', data = {'authkey': authkey,
'format': 'json'})
        postdata = r.json()
        user = postdata['username']
        DAG = postdata['data_access_group_name']

    if authkey!= '' and user != '':
        session['username'] = user
        session['DAG'] = DAG
        data = {'token': '373D231D597876BC6CFA0D479D40DA9E',
                'content': 'record',
                'format': 'json',
                'type': 'flat',
                'records[0]': '27',
                'records[1]': '28',
                'records[2]': '29',
                'records[3]': '30',
                'records[4]': '31',
                'records[5]': '32',
                'records[6]': '33',
                'records[7]': '34',
```

```
                'records[8]': '35',
                'records[9]': '36',
                'forms[0]': 'transplant_data',
                'rawOrLabel': 'raw',
                'rawOrLabelHeaders': 'raw',
                'exportCheckboxLabel': 'false',
                'exportSurveyFields': 'false',
                'exportDataAccessGroups': 'false',
                'returnFormat': 'json'
            }
        r = post('http://idipaz.org/api/', data = data)
        patients = r.json()
        if DAG != '':
            final_data = patients[patients.data_access_group == DAG]
        else:
            final_data = patients

        session['final_data'] = final_data

        return render_template('home.html')
    else:
        return 'Error: Please access through the bookmark on the REDCap project.'

@app.route('/home')
def home():
    return  render_template('home.html')

@app.route('/about')
def about():
    return  render_template('about.html')

@app.route('/plots/')
def plots():
    return  render_template('exploitation.html')

@app.route('/fig/lastyear')
def fig_lastyear():
    graphs.waitinglist()
    img = BytesIO()
    plt.savefig(img)
    plt.clf()
    img.seek(0)
    return send_file(img, mimetype='image/png')

@app.route('/plots/lastyear')
def plot_wl():
    return  render_template('plot_wl.html')
```

```
@app.route('/fig/historical')
def fig_historical():
    graphs.historical()
    img = BytesIO()
    plt.savefig(img)
    plt.clf()
    img.seek(0)
    return send_file(img, mimetype='image/png')

@app.route('/plots/historical')
def plot_ht():
    return  render_template('plot_ht.html')

@app.route('/fig/survival')
def fig_survival():
    graphs.survival()
    img = BytesIO()
    plt.savefig(img)
    plt.clf()
    img.seek(0)
    return send_file(img, mimetype='image/png')

@app.route('/plots/survival')
def plot_sv():
    return  render_template('plot_sv.html')

@app.route('/fig/followups')
def fig_followups():
    graphs.followups()
    img = BytesIO()
    plt.savefig(img)
    plt.clf()
    img.seek(0)
    return send_file(img, mimetype='image/png')

@app.route('/plots/followups')
def plot_fu():
    return  render_template('plot_fu.html')

if __name__ == "__main__":
    app.run()
```

config.py:

```
import os

class Config(object):
    SECRET_KEY = os.urandom(24)
```

```
class DevelopmentConfig(Config):
    DEBUG = True
    PORT = 5000
```

graphs.py:

```
def waitinglist():
    import matplotlib.pyplot as plt
    import matplotlib.patches as mpatches
    from matplotlib.collections import EventCollection
    import numpy as np

    # Fixing random state for reproducibility
    np.random.seed(19680)

    # create random data
    ydata = 500 * np.random.random([2, 12])

    # split the data into two parts
    waiting_list = ydata[1, :]
    transplanted = ydata[0, :]

    # sort the data so it makes clean curves
    waiting_list.sort()
    transplanted.sort()

    np.around(waiting_list)
    np.around(transplanted)

    # create some y data points
    xdata1 = ('Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun',
                        'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec')
    xdata2 = xdata1

    # plot the data
    fig = plt.figure()
    ax = fig.add_subplot(1, 1, 1)
    ax.plot(xdata1, waiting_list, 'r', xdata2, transplanted, 'b')

    red_patch = mpatches.Patch(color='red', label='Waiting list')
    blue_patch = mpatches.Patch(color='blue', label='Transplanted')
    plt.legend(handles=[red_patch, blue_patch], loc = 4)

    # create the events marking the y data points
    yevents1 = EventCollection(waiting_list, color=[1, 0, 0],
linelength=0.25,
                               orientation='vertical')
```

```
    yevents2 = EventCollection(transplanted, color=[0, 0, 1],
linelength=0.25,
                               orientation='vertical')

    # add the events to the axis
    ax.add_collection(yevents1)
    ax.add_collection(yevents2)

    # set the limits
    ax.set_ylim([0, 50])

    ax.set_title('Cumulative patients on the waiting list and
transplanted along the year')

    # display the plot
    plt.axis('tight')

def historical():
    import numpy as np
    import matplotlib.pyplot as plt

    np.random.seed(632)

    waitinglist = 450 + 200 * np.random.rand(1,5)
    waitinglist.sort()
    waitinglist = tuple(waitinglist.astype(int))
    waitinglist = (waitinglist[0][0], waitinglist[0][1],
waitinglist[0][2],
                   waitinglist[0][3], waitinglist[0][4])
    transplanted = 450 + 200 * np.random.rand(1,5)
    transplanted.sort()
    transplanted = tuple(transplanted.astype(int))
    transplanted = (transplanted[0][0], transplanted[0][1],
transplanted[0][2],
                   transplanted[0][3], transplanted[0][4])

    ind = np.arange(len(waitinglist))  # the x locations for the groups
    width = 0.35  # the width of the bars

    fig, ax = plt.subplots()
    rects1 = ax.bar(ind - width/2, waitinglist, width,
                    color='SkyBlue', label='Waiting list')
    rects2 = ax.bar(ind + width/2, transplanted, width,
                    color='IndianRed', label='Transplanted')

    # Add some text for labels, title and custom x-axis tick labels, etc.
    ax.set_ylabel('Number of patients')
    ax.set_title('Total patients on the waiting list and transplanted per
year')
```

```
    ax.set_xticks(ind)
    ax.set_xticklabels(('2014', '2015', '2016', '2017', '2018'))
    ax.legend()
    ax.axis(ymax = 701)


    def autolabel(rects, xpos='center'):
        """
        Attach a text label above each bar in *rects*, displaying its
height.

        *xpos* indicates which side to place the text w.r.t. the center
of
        the bar. It can be one of the following {'center', 'right',
'left'}.
        """

        xpos = xpos.lower()  # normalize the case of the parameter
        ha = {'center': 'center', 'right': 'left', 'left': 'right'}
        offset = {'center': 0.5, 'right': 0.175, 'left': 0.7}  # x_txt =
x + w*off

        for rect in rects:
            height = rect.get_height()
            ax.text(rect.get_x() + rect.get_width()*offset[xpos],
1.01*height,
                    '{}'.format(height), ha=ha[xpos], va='bottom')


    autolabel(rects1, "left")
    autolabel(rects2, "right")

def survival():
    import matplotlib.pyplot as plt
    import matplotlib.patches as mpatches
    from matplotlib.collections import EventCollection
    import numpy as np

    # Fixing random state for reproducibility

    np.random.seed(364)

    xdata1 = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
    xdata1 = np.array(xdata1)
    xdata2 = xdata1

    patientsurvival = [0] * 11
    patientsurvival[0] = 100
    for i in xdata1[1:11]:
```

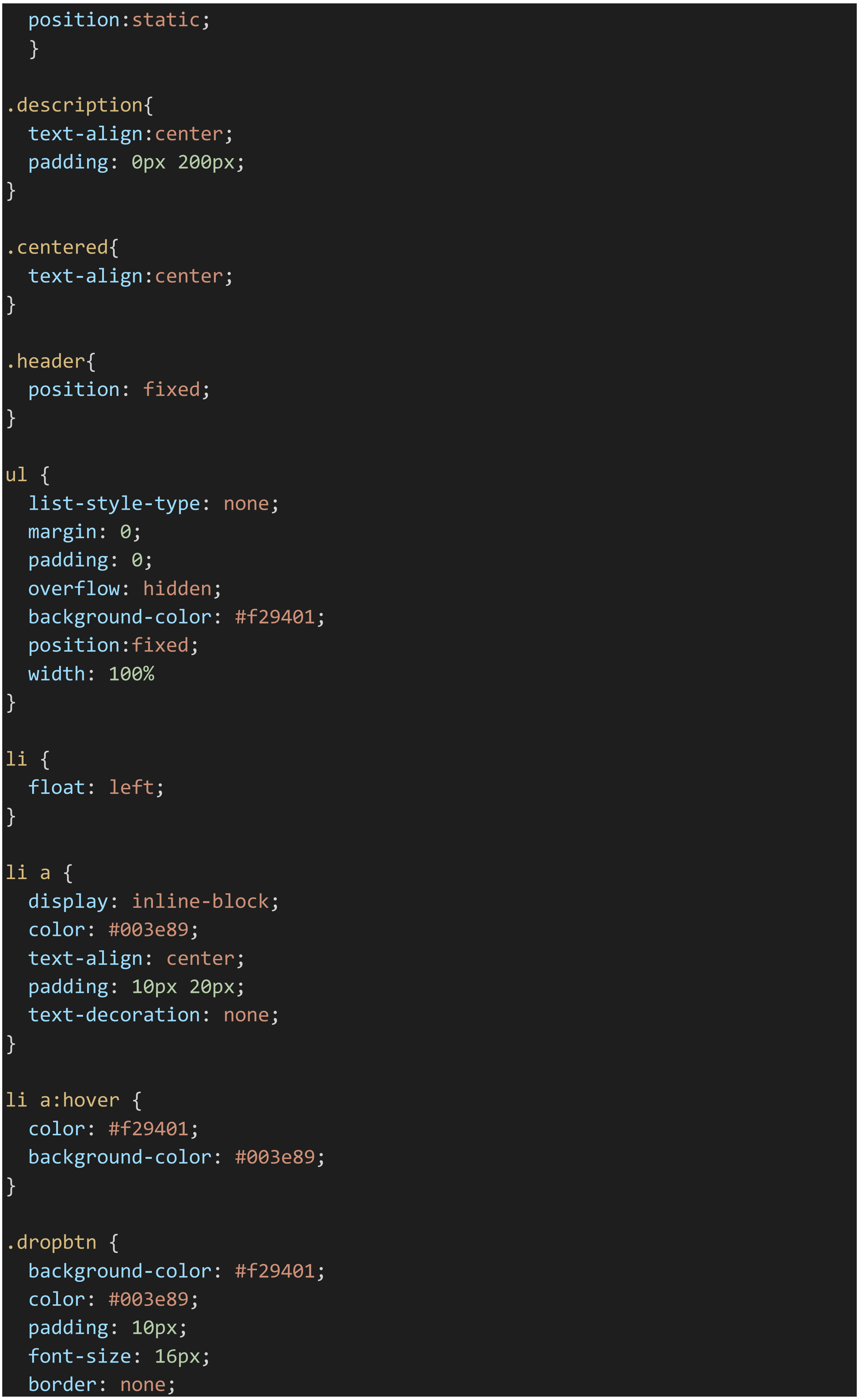
```
        val1 = np.random.rand() * 2.5 * xdata1[i]
        patientsurvival[i] =  patientsurvival[i-1] - val1

    graftsurvival = [0] * 11
    graftsurvival[0] = 100
    for i in xdata2[1:11]:
        val = np.random.rand() * 2.5 * xdata2[i]
        graftsurvival[i] =  graftsurvival[i-1] - val

    # plot the data
    fig = plt.figure()
    ax = fig.add_subplot(1, 1, 1)
    ax.plot(xdata1, patientsurvival, 'r', xdata2, graftsurvival, 'b')

    red_patch = mpatches.Patch(color='red', label='Patient survival')
    blue_patch = mpatches.Patch(color='blue', label='Graft survival')
    plt.legend(handles=[red_patch, blue_patch], loc = 3)

    # create the events marking the y data points
    yevents1 = EventCollection(patientsurvival, color=[1, 0, 0],
linelength=0.25,
                               orientation='vertical')
    yevents2 = EventCollection(graftsurvival, color=[0, 0, 1],
linelength=0.25,
                               orientation='vertical')

    # add the events to the axis
    ax.add_collection(yevents1)
    ax.add_collection(yevents2)

    # set the limits
    ax.set_ylim(0, 105)
    ax.set_xlabel('Years from transplant')
    ax.set_ylabel('Survival (in %)')

    plt.xticks(xdata1)

    ax.set_title('Percentage of patients and grafts survivals')

def followups():
    import numpy as np
    import matplotlib.pyplot as plt


    N = 4
    followups_alive = np.array([2096, 1733, 1304, 556])
    followups_dead = np.array([246, 161, 151, 347])
    followups_lost = np.array([102, 114, 138, 199])
    followups_opted_out = np.array([154, 88, 140, 202])
```

```

    ind = np.arange(N)    # the x locations for the groups
    width = 0.35       # the width of the bars: can also be len(x) sequence

    p1 = plt.bar(ind, followups_alive, width)
    p2 = plt.bar(ind, followups_dead, width,
                 bottom=followups_alive)
    p3 = plt.bar(ind, followups_lost, width,
                 bottom= followups_dead + followups_alive)
    p4 = plt.bar(ind, followups_opted_out, width,
                 bottom= followups_lost + followups_dead + followups_alive)

    plt.ylabel('Number of patients')
    plt.title('Total follow-ups made')
    plt.xticks(ind, ('1', '3', '5', '10'))
    plt.yticks(np.arange(0, 2601, 200))
    plt.legend((p1[0], p2[0], p3[0], p4[0]), ('Alive', 'Dead', 'Lost', 'Opted out'))
    plt.xlabel('Follow-up years')
```

main.css:

```
body{
  font-family: sans-serif;
  background: #eee0;
  margin: 0px
  }

a, h1, h2{
  color: #f29401
  }

h1, h2{
  font-family: 'Calibri',serif;
  margin: 0;
  padding-left: 20px
  }

h1{
  font-size:100px
  }

h2{
  border-bottom: 10px solid #003e89;
  font-size: 38px
```

```
  }

#logo{
  width: 500px;
  float: middle;
  height: 120px;
  padding-top: 5px;
  position: absolute;
}

.active {
  background-color: #0091d4 !important;
}

.text{
  text-align:justify;
  padding: 0px 200px;
  max-width: 50%;
  position:relative;
  top:60px;
  margin-left: auto;
  margin-right: auto;
}

#plot{
  display: block;
  width: 768px;
  height: 576px;
  position: static;
  margin-left: auto;
  margin-right: auto;
  padding:50px 0px 20px;
}

input{
  background-color: #f29401;
  color: #003e89;
  border: 2px solid #003e89;
  border-radius: 10px;
  margin-bottom: 40px;
  padding: 10px;
  cursor: pointer;
  box-shadow: 0 6px 6px 0 #003e89,
              0 6px 20px 0 rgba(0,0,0,0.19);
}

#introduction{
  text-align:center;
  padding: 50px 200px 0px;
```



```
  position:static;
  }

.description{
  text-align:center;
  padding: 0px 200px;
}

.centered{
  text-align:center;
}

.header{
  position: fixed;
}

ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #f29401;
  position:fixed;
  width: 100%
}

li {
  float: left;
}

li a {
  display: inline-block;
  color: #003e89;
  text-align: center;
  padding: 10px 20px;
  text-decoration: none;
}

li a:hover {
  color: #f29401;
  background-color: #003e89;
}

.dropbtn {
  background-color: #f29401;
  color: #003e89;
  padding: 10px;
  font-size: 16px;
  border: none;
```

```
  cursor: pointer;
}
.home{
  position: relative;
}

.others{
  position: relative;
  float: right;
}

.dropdown {
  position: relative;
  display: inline-block;
}

.dropdown-content {
  display: none;
  position: fixed;
  background-color: #003e89;
  min-width: 160px;
  box-shadow: 0px 8px 16px 0px #0091d4;
  z-index: 1;
}

.dropdown-content a {
  color: #f29401;
  background-color:#003e89;
  padding: 10px 10px;
  text-decoration: none;
  display: block;
}

.dropdown-content a:hover
{background-color: #0091d4
}

.dropdown:hover .dropdown-content {
  display: block;
}

.dropdown:hover .dropbtn {
  background-color: #003e89;
  color: #f29401
}
input:hover {
  color: #f29401;
  background-color: #003e89;
  }
```

home.html:

```
<html>
<head>

    <title>PETER: Home</title>
    <link rel="stylesheet" href="{{ url_for('static',
filename='css/main.css') }}" />
    <link rel="shortcut icon" href="{{ url_for('static',
filename='favicon.ico') }}">

    <h1>PETER
        <a href="https://www.transplantchild.eu/en/">
        <img src="{{url_for('static', filename='logo.png')}}"
alt="TransplantChild Logo" id="logo"/></a>
        <h2>PaEdiatric Transplantation European Registry</h2>
    </h1>

    <ul id="menu">
        <div class="home">
            <a href="/home"><li>Home</a></li>
        </div>
        <a href="/plots/"><div class="dropdown">
            <button class="active dropbtn">Data exploitation</button></a>
            <div class="dropdown-content">
                <a href="/plots/lastyear">Last year activity</a>
                <a href="/plots/historical">Historical activity</a>
                <a href="/plots/survival">Patient & Graft Survival</a>
                <a href="/plots/follow-ups">Follow-ups made</a>
            </div>
        </div>
        <div class="others">
            <li><a href="/about">About</a></li>
        </div>
    </ul>

</head>

<body>

    <p id="introduction">Welcome to the data exploitation website from
PETER.</p>
    <p class="description" >The patients' information is already
charged.To start the visualization, click on any of the options inside
the Data exploitation menu or on the following button.</p>

    <a href="/plots"><div class="centered"><input type="button"
value="Data exploitation"></div></a>
```

```
    <p class="description"> To go back to REDCap, click on the button below.</p>

    <a href="http://idipaz.org/redcap_v8.2.0/index.php?pid=41"><div class="centered"><input type="button" value="REDCap"></div></a>

</body>
</html>
```

exploitation.html:

```
<html>
<head>
    <title>PETER: Home</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='css/main.css') }}" />
    <link rel="shortcut icon" href="{{ url_for('static', filename='favicon.ico') }}">

    <h1>PETER<a href="https://www.transplantchild.eu/en/"><img src="{{url_for('static', filename='logo.png')}}" alt="TransplantChild Logo" id="logo"/></a><h2>PaEdiatric Transplantation European Registry</h2></h1>

    <ul id="menu">
        <div class="home">
            <a href="/home"><li>Home</a></li>
        </div>
        <a href="/plots/"><div class="dropdown">
            <button class="active dropbtn">Data exploitation</button></a>
            <div class="dropdown-content">
                <a href="/plots/lastyear">Last year activity</a>
                <a href="/plots/historical">Historical activity</a>
                <a href="/plots/survival">Patient & Graft Survival</a>
                <a href="/plots/follow-ups">Follow-ups made</a>
            </div>
        </div>
        <div class="others">
            <li><a href="/about">About</a></li>
        </div>
    </ul>


</head>
<body>

    <p id="introduction">Welcome to the Data exploitation section. Here you can find the different plots offered for data analysis:</p>
```

```
    <p class="description" >1. Last year activity: It represents, month
by month, the cumulative number of patients registered on the waiting
list versus the total patients transplanted during the same period of the
year.</p>

    <a href="/plots/lastyear"><div class="centered"><input type="button"
value="Last year activity"></div></a>

    <p class="description" >2. Historical activity: It displays the total
number of patients transplanted and on the waiting list at the end of
every previous year.</p>

    <a href="/plots/historical"><div class="centered"><input
type="button" value="Historical activity"></div></a>

    <p class="description" >3. Patient & Graft Survival: It contains, in
percentage, the survivability of both patient and graft until 10 years
after the transplant.</p>

    <a href="/plots/survival"><div class="centered"><input type="button"
value="Patient & Graft Survival"></div></a>

    <p class="description" >4. Follow-ups made: In this plot, the follow-
ups are registered per result (Alive, Dead, Lost or Opted out) and per
time from transplant (1, 3, 5 or 10 years).</p>

    <a href="/plots/follow-ups"><div class="centered"><input
type="button" value="Follow-ups made"></div></a>

</body>
</html>
```

plot_wl.html (only the template for one of the plot URLs is included, as they only change some descriptions and the own plot):

```
<html>
<head>
  <title>PETER: Waiting list vs Transplanted patients last year</title>
  <link rel="stylesheet" href="{{ url_for('static',
filename='css/main.css') }}" />

  <link rel="shortcut icon" href="{{ url_for('static',
filename='favicon.ico') }}">

  <h1>PETER<a href="https://www.transplantchild.eu/en/"><img
src="{{url_for('static', filename='logo.png')}}" alt="TransplantChild
```

```html
Logo" id="logo"/></a><h2>PaEdiatric Transplantation European
Registry</h2></h1>

  <ul id="menu">
    <div class="home">
      <a href="/home"><li>Home</a></li>
    </div>
    <a href="/plots/"><div class="dropdown">
      <button class="active dropbtn">Data exploitation</button></a>
      <div class="dropdown-content">
        <a href="/plots/lastyear">Last year activity</a>
        <a href="/plots/historical">Historical activity</a>
        <a href="/plots/survival">Patient & Graft Survival</a>
        <a href="/plots/follow-ups">Follow-ups made</a>
      </div>
    </div>
    <div class="others">
      <li><a href="/about">About</a></li>
    </div>
  </ul>
</head>
<body>

  <img src="{{ url_for('fig_lastyear', lastyear = title) }}" id="plot"
alt="Comparison of Waiting list vs Transplanted patients in the previous
year." />

  <div class="centered"><input type="button" value="Apply filters"> &nbsp
&nbsp &nbsp<a href="/plots/historical"><input type="button"
value="Continue to Historical activity"></a></div>

</body>
</html>
```

about.html:

```html
<html>
<head>

  <title>PETER: About us</title>

  <link rel="stylesheet" href="{{ url_for('static',
filename='css/main.css') }}" />

  <link rel="shortcut icon" href="{{ url_for('static',
filename='favicon.ico') }}">
```

```html
  <h1>PETER<a href="https://www.transplantchild.eu/en/"><img
src="{{url_for('static', filename='logo.png')}}" alt="TransplantChild
Logo" id="logo"/></a><h2>PaEdiatric Transplantation European
Registry</h2></h1>

  <ul id="menu">
    <div class="home">
      <a href="/home"><li>Home</a></li>
    </div>
    <a href="/plots/"><div class="dropdown">
      <button class="dropbtn">Data exploitation</button></a>
      <div class="dropdown-content">
        <a href="/plots/lastyear">Last year activity</a>
        <a href="/plots/historical">Historical activity</a>
        <a href="/plots/survival">Patient & Graft Survival</a>
        <a href="/plots/follow-ups">Follow-ups made</a>
      </div>
    </div>
    <div class="active others">
      <li><a href="/about">About</a></li>
    </div>
  </ul>
</head>
<body>
<div class="text"><p>PETER (PaEdiatric Transplantation European Registry)
is a new rational, efficient, interoperable, open, accessible and
predictive paediatric transplant registry: the first that includes
jointly solid organ transplantation (SOT) and hematopoietic stem cell
transplantation (HSCT). PETER is based on a crosscutting approach
regardless of the type of transplant, integrating variables from the
different phases. It enables increase of knowledge on this rare medical
condition in children and development of clinical research with the main
objective of improved survival and quality of life for paediatric
patients of EU and their families.  PETER leans on an interoperable
technology platform with a module of business intelligent that allows it
to be a predictive registry directly impacting on the (a) process of
patient care, improving it, (b) transplanted patients and families
decreasing the morbidity and mortality, (c) health care professional and
institutions assisting in the process of personalised clinical decision-
making, (d) public health systems and sanitary authorities increasing the
power of analysis in terms of distribution of capacities and
resources.</p>
<p>If any problem is encountered when utilizing this platform please
contact gonzalo.sofio@idipaz.es or
coordination@transplantchild.eu</p></div>

</body>
</html>
```