Leonardo Pellizzoni

PROCESSO DE MONITORAÇÃO DE NOVAS TECNOLOGIAS EM SAÚDE

Dissertação apresentada à Universidade de Caxias do Sul, para obtenção do título de Mestre em Ciências da Saúde.

Caxias do Sul

2019

Leonardo Pellizzoni

PROCESSO DE MONITORAÇÃO DE NOVAS TECNOLOGIAS EM SAÚDE

Dissertação apresentada à Universidade de Caxias do Sul, para obtenção do título de Mestre em Ciências da Saúde.

Orientador: Dr. Asdrubal Falavigna

Co-Orientador: Drª. Scheila de Avila e Silva

Caxias do Sul

2019

Dados Internacionais de Catalogação na Publicação (CIP)
Universidade de Caxias do Sul
Sistema de Bibliotecas UCS - Processamento Técnico

P391p Pellizzoni, Leonardo
Processo de monitoração de novas tecnologias em saúde / Leonardo Pellizzoni. – 2019.
56 f. : il. ; 30 cm

Dissertação (Mestrado) - Universidade de Caxias do Sul, Programa de Pós-Graduação em Ciências da Saúde, 2019.
Orientação: Asdrubal Falavigna.
Coorientação: Scheila de Avila e Silva.

1. Tecnologia médica - Avaliação. 2. Registros médicos. 3. Qualidade de vida. I. Falavigna, Asdrubal, orient. II. Silva, Scheila de Avila e, coorient. III. Título.

CDU 2. ed.: 61:004

Catalogação na fonte elaborada pela(o) bibliotecária(o)
Michele Fernanda Silveira da Silveira - CRB 10/2334

UNIVERSIDADE DE CAXIAS DO SUL

Programa de Pós-graduação em Ciências da Saúde

COORDENADOR DO PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA SAÚDE

PROF. DR. ASDRUBAL FALAVIGNA

PROCESSO DE MONITORAÇÃO DE NOVAS TECNOLOGIAS EM SAÚDE

Leonardo Pellizzoni

Dissertação de Mestrado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Ciências da Saúde da Universidade de Caxias do Sul, como parte dos requisitos necessários para a obtenção do título de Mestre em Ciências da Saúde, Linha de Pesquisa: Investigação clínica e epidemiológica

Caxias do Sul, 29 de julho de 2019.

Banca Examinadora:

Dra. Ricardo Vieira Botelho
IAMSPE

Dr. Leandro Luís Corso
UCS

Dr. Sandro Rogério dos Santos
UCS

Dr. Asdrubal Falavigna
UCS
Orientador

Dedicatória

Para minha esposa Francieli, pilar da minha vida.

Agradecimentos

Ao meu pai **Juarez** (*in memorian*) pelo exemplo e aprendizado que se mantem presentes.

Ao **Dr. Asdrubal Falavigna** pelo seu tempo, paciência e amizade que me proporcionaram aprendizado extraordinário e com quem tive enorme prazer em trabalhar.

A **Drª. Scheila de Avila e Silva** pela ajuda, conselhos e incentivos desde a minha graduação.

Ao **Frederico Criscuoli** juntamente com os estudantes do curso de medicina e o **Vicenzo Falavigna** pela auxilio, testes e sugestões ao processo de monitoração.

Aos professores e colegas do mestrado pelo companheirismo e aprendizado. Em especial aos queridos amigos **Bruno Saciloto**, **Charles André Carazzo** e **Leonardo do Nascimento**.

A competente **Haiany Serraggio de Souza** por toda ajuda e auxílio.

A **Iraci Cristina de Carli** e **Alexandre Nascimento** pelo encorajamento em seguir estudando e se aperfeiçoando mesmo que em áreas não diretamente relacionadas a computação.

A toda minha **família** pelo suporte.

A minha esposa **Francieli** pela enorme paciência, incentivo, ajuda e compreensão durante mais uma etapa importante da minha vida.

Sumário

Esta dissertação de Mestrado Acadêmico Stricto Sensu é apresentada no formato exigido pelo Programa de Pós-Graduação em Ciências da Saúde da Universidade de Caxias do Sul. A mesma é constituída da secção de “Introdução com referências bibliográficas”, a inclusão do artigo original submetido/publicado em periódico Qualis A na classificação da Coordenação de Aperfeiçoamento de Pessoal em Nível Superior (CAPES), e as “Considerações Finais e Perspectivas”.

1 INTRODUÇÃO

A avaliação de tecnologias em saúde (ATS) por meio de um conjunto de dados permite a comparação de diferentes tratamentos clínicos em múltiplas áreas da medicina, além de usar eficientemente os recursos [1,2]. Uma das formas de criar-se um conjunto de dados é por meio de questionários. Dentre estes questionários os patient-reported outcomes (PRO) distinguem-se por coletarem dados diretamente do paciente sem a necessidade de terceiros ou intervenção médica [3].

Os dados coletados através de questionários são utilizados para auxiliar na tomada de decisão, mensurar critérios de eficácia, acurácia e efetividade da atividade diária do profissional da saúde e de qualidade de vida dos pacientes [4,5]. Além aos benefícios já mencionados o conjunto de informações agrega valor estratégico a prática médica por diminuir a distância entre as visões do paciente e do médico, bem como possibilitar um tratamento de acordo com as necessidades especificas [6].

A coleta online de dados é um fator importante e possibilita que os pacientes se mantenham engajados em continuar a responder questionários quando a utilização destas informações é percebida como útil por eles [7]. No entanto existem algumas barreiras que impedem a utilização desta forma de coleta dos dados: falta de tempo, falta de assistência na coleta dos dados, longo tempo de preenchimento dos questionários pelos pacientes e a falta de um banco de dados [8].

A utilização de sistemas para auxiliar estes e outros aspectos de modo a acompanhar e avaliar tratamentos clínicos já é documentada e utilizada [9]. A coleta de dados dos pacientes pode ser conduzida através de sistemas criados com o objetivo de atender especificamente determinadas práticas clínicas e questionários previamente definidos pelos pesquisadores [10–12]. Neste tipo de abordagem, o sistema é criado especificamente para as necessidades de um estudo clínico em particular. Isto deve-se ao fato de que os instrumentos de medição podem variar de acordo com o tipo da população alvo, o propósito de cada estudo clínico e o que se pretende medir [13,14].

A atividade diária de médicos e pesquisadores prescinde da informação da qualidade do trabalho ofertado, sendo necessário o uso de questionários de acordo com populações, práticas clínicas e o que se pretende medir [15]. Fica visível a necessidade de plataformas digitais que permitam uma flexibilização e facilidade na criação do banco de dados. Esta necessidade é parcialmente atendida através de sistemas específicos da área da saúde, como o RedCap [16] e EpiData [17]. De maneira mais genérica esse também é o foco de ferramentas como Google Forms ou SurveyMonkey. Estes métodos citados facilitam o processo, entretanto vem associado com custos, mão de obra para a correta execução dos bancos de dados e da ausência de funcionalidades das ferramentas [18]. Considerando as vantagens da ATS elencadas previamente, as dificuldades impostas na utilização de instrumentos de medição e a necessidade de flexibilizar os questionários usados de acordo com as características do estudo clínico e população, este trabalho tem como objetivo automatizar e auxiliar o processo de coleta de informações para análise clínica permitindo o uso dos dados em tempo real.

Para atender o objetivo do trabalho foi construído um sistema que permite realizar a coleta dos dados diretamente pelos próprios pacientes, mantendo as informações em bancos de dados para utilização em tempo real na visualização de gráficos e envio para software de análises estatísticas. A arquitetura de software que suporta o sistema construído é explicada no primeiro artigo deste trabalho, sendo que a estrutura poderá ser utilizada na construção de outros sistemas, disponível na página 13. Os detalhes da criação do sistema, do processo e seu funcionamento para a coleta de dados estão definidos no segundo artigo deste trabalho, disponível na página 30.

2 REFERÊNCIAS

1. Ministério da Saúde. Secretaria de Ciência T e IED de C e T. Avaliação de Tecnologias em Saúde: institucionalização das ações no Ministério da Saúde. Rev Saúde Pública. 2006 Aug;40(4):743–7.

2. Drummond MF, Schwartz JS, Jönsson B, Luce BR, Neumann PJ, Siebert U, et al. Key principles for the improved conduct of health technology assessments for resource allocation decisions. Int J Technol Assess Health Care. 2008 Jul 4;24(03):244–58.

3. Spertus J. Barriers to the Use of Patient-Reported Outcomes in Clinical Care. Circ Cardiovasc Qual Outcomes. 2014 Jan 1;7(1):2–4.

4. Falavigna A, Teles AR, Braga GL de, Barazzetti DO, Lazzaretti L, Tregnago AC. Instrumentos de avaliação clínica e funcional em cirurgia da coluna vertebral. Coluna/Columna. 2011;10(1):62–7.

5. Falavigna A, Righesso Neto O, Teles AR. Avaliação clínica e funcional no pré-operatório de doenças degenerativas da coluna vertebral. Coluna/Columna. 2009 Sep;8(3):245–53.

6. Falavigna A, Dozza DC, Teles AR, Wong CC, Barbagallo G, Brodke D, et al. Current Status of Worldwide Use of Patient-Reported Outcome Measures (PROMs) in Spine Care. World Neurosurg. 2017 Dec;108:328–35.

7. Santana MJ, Haverman L, Absolom K, Takeuchi E, Feeny D, Grootenhuis M, et al. Training clinicians in how to use patient-reported outcome measures in routine clinical practice. Qual Life Res. 2015 Jul 15;24(7):1707–18.

8. Teles AR, Righesso O, Gullo MCR, Ghogawala Z, Falavigna A. Perspective of Value-Based Management of Spinal Disorders in Brazil. World Neurosurg. 2016 Mar 1;87:346–54.

9. Goeree R, He J, O'Reilly D, Tarride J-E, Xie F, Lim M, et al. Transferability of health technology assessments and economic evaluations: a systematic review of approaches for assessment and application. Clinicoecon Outcomes Res. 2011;3:89–104.

10. Asher AL, Speroff T, Dittus RS, Parker SL, Davies JM, Selden N, et al. The National Neurosurgery Quality and Outcomes Database (N2QOD). Spine (Phila Pa 1976). 2014 Oct 15;39(22 Suppl 1):S106–16.

11. Crandall W, Kappelman MD, Colletti RB, Leibowitz I, Grunow JE, Ali S, et al. ImproveCareNow: The development of a pediatric inflammatory bowel disease improvement network. Inflamm Bowel Dis. 2011 Jan;17(1):450–7.

12. Marinelli M, Positano V, Lorenzoni V, Caselli C, Mangione M, Marcheschi P, et al. A modular informatics platform for effective support of collaborative and multicenter studies in cardiology. Health Informatics J. 2016 Dec 26;22(4):1083–100.

13. Godil SS, Parker SL, Zuckerman SL, Mendenhall SK, McGirt MJ. Accurately measuring the quality and effectiveness of cervical spine surgery in registry efforts: determining the most valid and responsive instruments. Spine J. 2015 Jun 1;15(6):1203–9.

14. Prinsen CAC, Vohra S, Rose MR, Boers M, Tugwell P, Clarke M, et al. How to select outcome measurement instruments for outcomes included in a “Core Outcome Set” – a practical guideline. Trials. 2016 Dec 13;17(1):449.

15. Nelson EC, Eftimovska E, Lind C, Hager A, Wasson JH, Lindblad S. Patient reported outcome measures in practice. BMJ. 2015 Feb 10;350:g7818.

16. Harris PA, Taylor R, Thielke R, Payne J, Gonzalez N, Conde JG. Research electronic data capture (REDCap)--a metadata-driven methodology and workflow process for providing translational research informatics support. J Biomed Inform. 2009 Apr;42(2):377–81.

17. Singh S. Review of epidata entry and analysis freewares. Indian J Community Med. 2009 Jan;34(1):76–7.

18. McGirt MJ, Parker SL, Asher AL, Norvell D, Sherry N, Devin CJ. Role of Prospective Registries in Defining the Value and Effectiveness of Spine Care. Spine (Phila Pa 1976). 2014 Oct 15;39(22 Suppl 1):S117–28.

3 ARTIGO SOBRE ARQUITETURA DE SOFTWARE DO PROCESSO

METHODOLOGY TO REUSE, DECOUPLE AND ISOLATE BUSINESS RULES IN SOFTWARE ARCHITECTURE

LEONARDO PELLIZZONI
Health Sciences Postgraduate Program, University of Caxias do Sul (UCS), Brazil
lpellizzoni@ucs.br
ALEXANDRE E. K. NASCIMENTO
University of Caxias do Sul (UCS), Brazil
aeknasci@ucs.br
SCHEILA A. SILVA
University of Caxias do Sul (UCS), Brazil
sasilva6@ucs.br
ASDRUBAL FALAVIGNA
Health Sciences Postgraduate Program, University of Caxias do Sul (UCS), Brazil
afalavig@ucs.br

Abstract: Systems embody domain knowledge originating from stakeholders which is implemented as business rules and specifies how the system works. Business rules exist independent of procedures, workflows and technologies, and also tend to remain more stable than the technologies used to carry out the operations performed by users in many user interfaces. A model to reuse the business rules with a high level of decoupling is proposed and applied in a layered software architecture. The findings support systems evolution and adaptations of continuously evolving technologies or new interfaces available to interact with users.

Keywords: Business logic; business rules; reuse; software architecture; software engineering; systems evolution.

1 Introduction

Stakeholders provide the business rules or business logic in the software development process, i.e., the specification of business domain knowledge. In their simplest form, business rules can be defined as a part of the system which specifies its basic functionalities. This information is an important asset of the system and adds further value by delimiting where in the system organization they were defined [1], [2]. Some fundamental principles should guide the implementation of business rules such as: (*i)* they should be explicit, single-sourced and easily manageable; (*ii*) they can exist independent of procedures, workflows and technologies, (*iii*) they should present a high level of decoupling. This last one is a key feature because it divides the software into independent parts or modules, consequently decreasing the impact of future changes and errors [3], [4].

The business rules of systems can be implemented with design patterns such as domain and services or even in multiple layers of the software architecture, i.e., user interface and persistence layer [5]–[7]. Software architecture is the fundamental organization of a system, the relations among their components and the principles guiding its design and evolution [8].There are several styles and methods of software architecture, such as pipeline and filters, layered systems, event-based, implicit invocation and domain-specific software architectures. A layer-based software architecture is composed by different levels, specified as layers, each one dedicated to a specific part of the system. From the systems life cycle perspective a layered architecture promotes the creation and evolution of parts of the system independently, rendering them more portable, easily changed and reusable. The use of architectural techniques adds strategic elements to the software development, minimizing costs, potentiating reuse and increasing the amount of system functionality over time [9]–[13].

The common layers used in software architecture are presentation, domain and data source. The domain layer maintains the business rules relevant for the system, which are expressed in source code and tend to keep stable for longer than the technologies used to access them [7], [14], [15]. For this reason the higher the level of decoupling, interoperability and portability, the more value is added to the system since it will be possible to incorporate new technologies, or even reuse the business core in future systems [16].

This paper was motivated by the connection between business rules and the evolving technologies necessary to make them available to users in the long-term. This configure an important topic in software engineering since business rules tend to be more stable than technologies over time [14] and software systems are dependent to their architecture to ensure long-term use, efficient maintenance, and appropriate evolution in a continually changing environment [17]. The main goal is to describe the methodology used to decouple and reuse the business rules and how they are organized in the software architecture, independently of the user interface and the technology that stores data.

2 Proposed Model

This paper considered two main assumptions to reuse the business rules. The first was the possibility of creating a new presentation and the persistence layers reusing the business rules. The second assumption was that no specific technology be imposed to interact with users or save information in the database, delegating these choices according to the requirements of the system. The proposed model in a layer software architecture is presented in Figure 1.

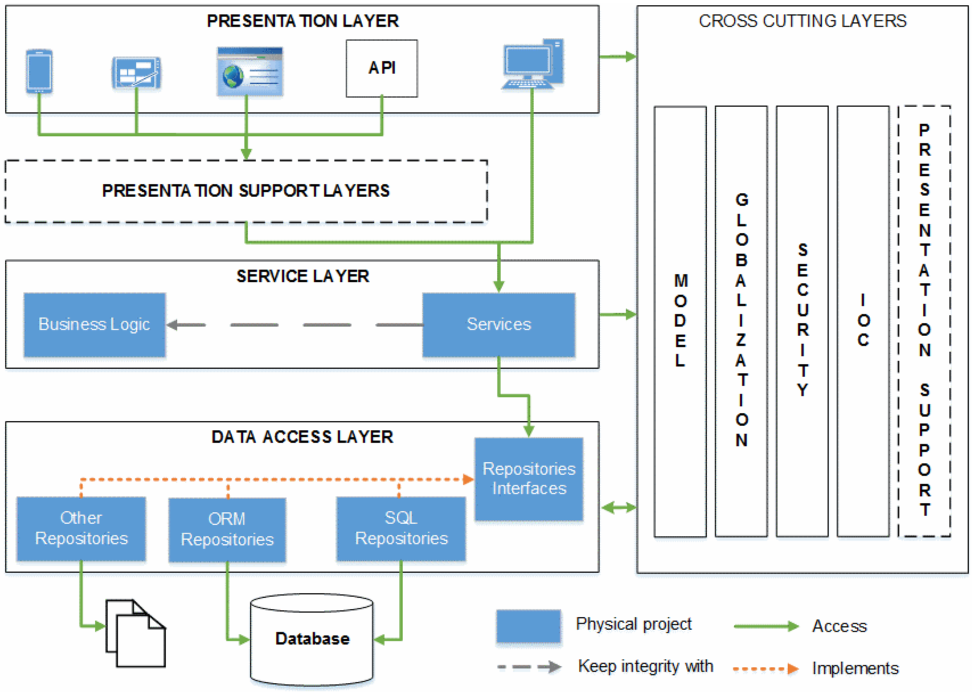


Figure 1 Software architecture diagram

The communication between layers specifies that a given layer only accesses the layer immediately below it or the crosscutting ones [18]. One specific layer interacts with the other by method or functions which will return a value or present an exception which characterizes the response back to the top layer without knowing which one it was. The cross-cutting layers provide the necessary support for the systems features and even other layers with technical support [19].

The software architecture diagram (Figure 1) shows the Globalization and Security layers. These layers are optional and will be used according to the system requirements. The Globalization layer is used when an application must interact with users in multiple languages [20]. The globalized content related to the system is stored in the database or resource file, and the layer provides the correct text according to the user's nationality. All other operations are delegated to the programming language [21], [22]. The security layer provides system safety by controlling access and operations performed in the system [18]. The first issue of security is resolved by login mechanisms that create a user profile, which comprises the operations allowed for it. The second issue of security is resolved by multiple profiles created and managed by the administrator and each one has a customized list of permissions for the users, which ensures control of the operations performed. The IOC layer is established in the cross-cutting section of the architecture. It is responsible for implementing the dependency injection pattern through libraries or factory patterns, which support other layers to promote decoupling by associating a concrete class with an interface [7]. This layer is essential for the decoupled implementation of data access and globalization layers.

2.1 Model layer

The Model layer reflects the behavior of the domain pattern to define entities, however it does not implement any business logic [6], [23]. This layer is composed by entities that represent value for the business, for instance: client, order, person, among others. The implementation of entities in this layer must use primitive types or other classes previously created in this layer. This premise of using primitive types together with isolation of the business rules increases the interoperability and reuse criteria of this software architecture [4], [24].

2.2 Service layer

Considering the previously mentioned assumptions the proposed architecture focuses on the service and model layers. The source code implemented in the service layer is based on the service pattern, and as a result the services created reflect the system's features [6], [7], [25]. Typically, one service is associated with at least one model entity and represents the business operations related to them. The service layers in this architecture separate the business rules from the service itself in a logical and physical form. With this separation there is the benefit of explicitly distinguishing one from another, and also potentiating the business rules to be high testable, reusing them over the long term and enforcing the interoperability with the assets of the system [26]. Every service created is available to be used by superior layers and enforce the business knowledge through the business rules. This structure potentiated two ways of creating automated tests, which will ensure the quality of the system in the maintenance and evolution of software. The first is to directly test the business rules by simulating an operation related to a specific part of that business knowledge associated with an entity. The second form is by testing the service and embracing all the logic related to a specific feature, because it could use various operations, and besides it is the same path that will be accessed by end-users.

2.3 Data access layer

The data access layer is responsible for making the connection between the database and the technology used to access the data. Typically, these technologies are Object-Relational Mapping (ORM) frameworks or pure Structured Query Language (SQL) instructions [27]. The purpose of this layer, in the present architecture, is to create a logical and physical separation between the interfaces and their implementation. The operation required by the service layer is carried out by interfaces, which are associated with a concrete class by the Inversion of Control (IOC) layer. The benefits of this separation are based on the easy modification of the database and the technology that accesses them, as well the decoupling of the operations and technology.

3 Illustrating the Model

The proposed model was used to develop two different systems using the ICONIX method [28]. The first software (S1) is dedicated to applying patient reported outcome (PRO) questionnaires with the intention of collecting data directly from the patients. The second software (S2) is a system that stores and manages information related to a group of people. The technological differences originated from the requirements of each software are detailed in Table 1 and the details of S1 will be submitted elsewhere.

	Software 1 (S1)	**Software 2 (S2)**
Database	SQL Server	PostgreSQL
Presentations	Web (Asp.Net) for interaction with final users.	Desktop (Windows Forms) for interaction with final users.
	Console used as WebJob to automate notification tasks.	Web (Asp.Net) for interaction with final users.
Data access implementation	ORM (Entity Framework)	SQL Instructions (via Npgsql library)
		ORM (Entity Framework)

Table 1: Technological differences between the systems developed with the same software architecture

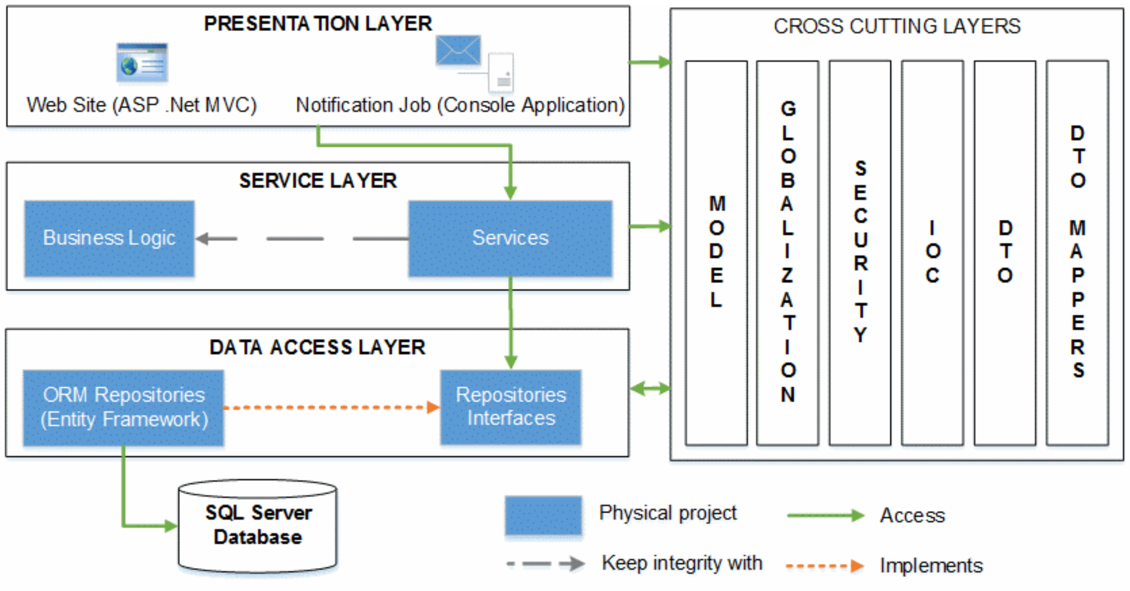


Figure 2: S1 Software architecture

According to the system requirements, different architectures were defined for each one as presented in Figure 2 and Figure 3. For instance, the globalization and security layer were used only in S1, since it was required to manage permissions for access and the interaction with users in English, Portuguese and Spanish. The communications between layers were handled with custom exceptions created to enforce the communication from bottom layers to the top ones. The presentation layer implements the correct treatment to display exceptions in a user-friendly format. It is important to point out that there is a different treatment for the custom exceptions and the languages exceptions, both are treated properly but the custom exceptions contain relevant business information. Apart from the exceptions, the communication between layers occurs by functions or methods.

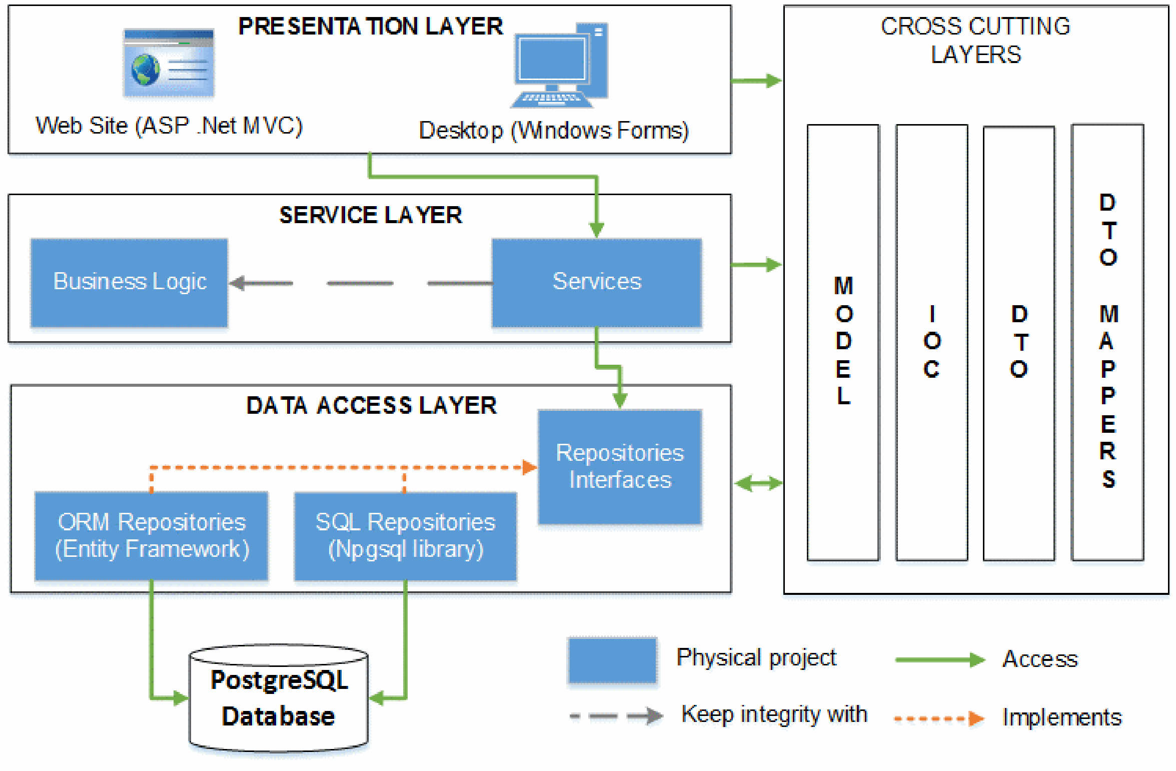


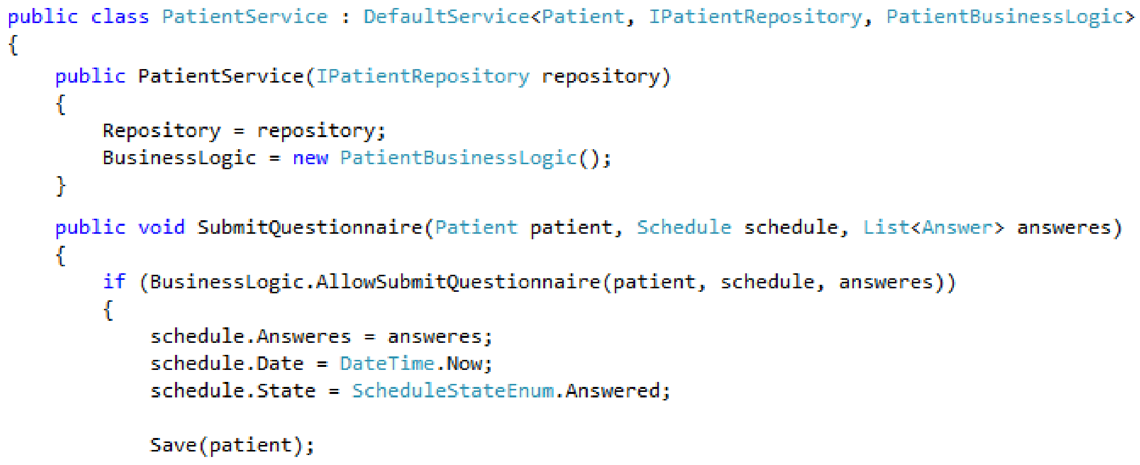
Figure 3: S2 Software architecture

The objects created in the model layer only use primitive types and a composition of other objects previously created in the same layer. All the classes created represent only the structure that stores the data for the associated business entity without any validation. The requirement of an S1 system forces the use of globalization and security mechanisms, both implemented as cross-cutting layers. The globalization layer was divided into two projects; the main one contains the class used by the other layers to obtain a content in a given language. The other project implements the necessary interfaces in a class that accesses the content storage in resource files for each language [21]. The links between interface and class are in the IOC layer, which potentiates other mechanisms to store the content without refactoring the systems. The other concerns related to the culture such as formats were delegated to the .Net Framework.

Security was implemented though the login and authentication mechanisms available on the ASP .Net as well as a custom attribute that was inherited from AuthorizeAttribute [21]. The custom attribute was placed in every action of the Controller that has a corresponding operation in the custom profile created by the administrator. Both systems were implemented in C# and most of the projects created were Class Libraries projects in which the output is a Dynamic Link Library (DLL). Exceptions were made only in the presentation layer, wherein the type of project is related to the specific technology used. If the programming language was Java, the equivalent to Class Library projects could be the package with Java Archive (JAR) output. All activities necessary to create the source code structure and implemented requirements were mapped into processes after the prototyping phase of the software architecture. Those processes were executed in both systems and they are available for other systems only for this architecture.

3.1 Service layer

All the features in the system have at least one service, which will implement the necessary operations for the associated entity. The most common operations are creating, updating, retrieving and deleting, besides those exclusive to the entity. A fraction of a service implemented for S1 software is visualized in Figure 4 and the respective business rules in Figure 5.



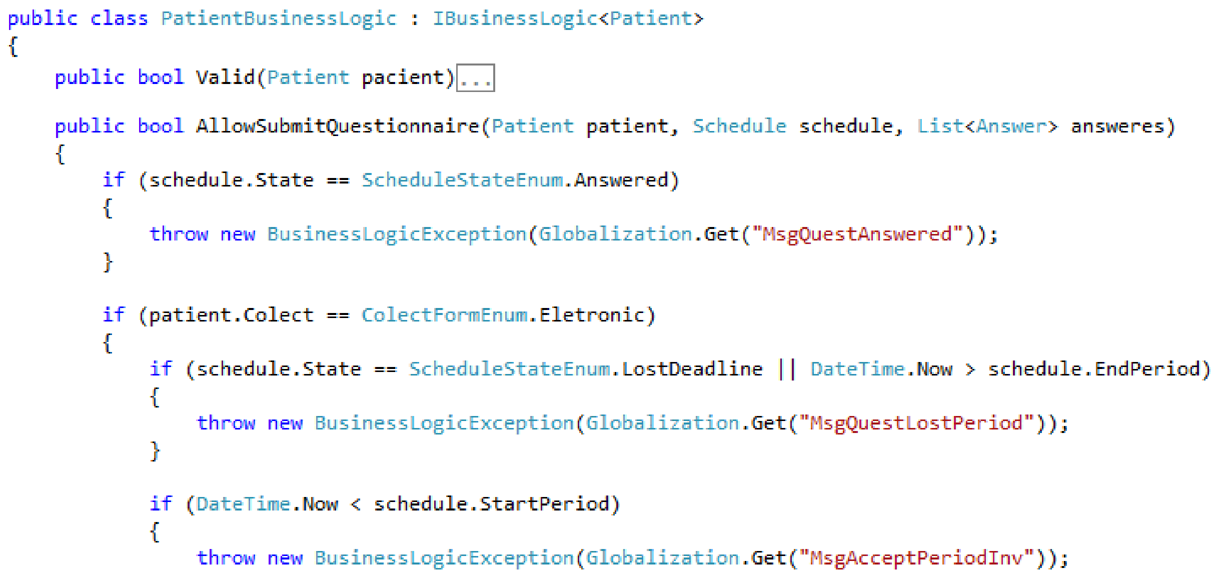
```
public class PatientService : DefaultService<Patient, IPatientRepository, PatientBusinessLogic>
{
    public PatientService(IPatientRepository repository)
    {
        Repository = repository;
        BusinessLogic = new PatientBusinessLogic();
    }

    public void SubmitQuestionnaire(Patient patient, Schedule schedule, List<Answer> answeres)
    {
        if (BusinessLogic.AllowSubmitQuestionnaire(patient, schedule, answeres))
        {
            schedule.Answeres = answeres;
            schedule.Date = DateTime.Now;
            schedule.State = ScheduleStateEnum.Answered;

            Save(patient);
```

Figure 4: Example of a service implemented in the S1 software

The service uses the repository interface defined in the data access layer and the business rules class which are linked to the entity represented by that service. The repository interface is employed instead of a concrete class to decouple the service from the technology and the mechanism used to store and access data. In addition, it also provides a simple way to perform an automated test of the services, for example, implementing a memory repository for the interface. With the amount of services developed it was possible to create one class defined as “DefaultService” for the purpose of reusing the common code and behavior for all services. This class is available in a separate library deployed by the NuGet package Manager.



```
public class PatientBusinessLogic : IBusinessLogic<Patient>
{
    public bool Valid(Patient pacient)...

    public bool AllowSubmitQuestionnaire(Patient patient, Schedule schedule, List<Answer> answeres)
    {
        if (schedule.State == ScheduleStateEnum.Answered)
        {
            throw new BusinessLogicException(Globalization.Get("MsgQuestAnswered"));
        }

        if (patient.Colect == ColectFormEnum.Eletronic)
        {
            if (schedule.State == ScheduleStateEnum.LostDeadline || DateTime.Now > schedule.EndPeriod)
            {
                throw new BusinessLogicException(Globalization.Get("MsgQuestLostPeriod"));
            }

            if (DateTime.Now < schedule.StartPeriod)
            {
                throw new BusinessLogicException(Globalization.Get("MsgAcceptPeriodInv"));
```

Figure 5: Example of the business validation implementation in a business rules class

The business rules maintain the functional and non-functional business requirements related to a model and its dependencies in a class implemented as functions or methods. In some implemented business rules it is necessary to use the repository as a third party to execute the validation properly. In the present cases the repository interface was used on the class to obtain the data and therefore conclude the process.

3.2 Data access layer

For each system was created a project that maintains the interfaces for all the features and defines the available operations to be performed in the database. The Entity Framework version 6.1.3 was used in the S1 and S2 software, each with the respective project and implementing all the interfaces defined previously. The S2 also implemented the database access through SQL commands with the Npgsql library version 2.2.7. The S2 software has two data access layers to validate the change of layers because it has less features than S1 to be reimplemented.

As a result of not attaching the Table and Column attributes [37] to the model classes, it was necessary to create a mapper between the class with the table and the properties with the columns. This process was executed for all the tables in the database that have a corresponding model class, and then added to the model builder of the Entity Framework DbContext. The mappers created the EntityTypeConfiguration [21] class and are available only in the data access layer.

The link between interfaces and classes was resolved with a container implemented using the StructureMap library version 4.4.0 which. Other layers used the container to pass an interface and obtain the respective class created by the library itself. The StructureMap uses the registry files to know what classes have to be created for that interface, as well the lifecycle of that object [29]. Each system (S1 and S2) implemented this layer with one container and the registry files. When it is necessary to change the technology used in a layer, the modification is performed in the registry file, by changing the class associated with the interface without refactoring the other layers.

4 Related Work

The approach of treating business rules separately from the systems is addressed by storing them in Extensible Markup Language (XML) files [30]. Although XML is a technique that ensures interoperability it would be necessary to have a middle part to access and execute the business rules. In addition, rule management would be more complex and could lose the programming language’s potential. Other way to treat business rules is by developing a method which focuses on management and reuses business rules considering their evolution [5], or by creating a topology that considers business rules the most volatile part of the systems [31]. The results obtained using those methods do not specify the separation and isolation which improves the achievement of a high level of decoupling that is a key point of this work.

Other papers also developed new architectures to achieve the reuse of business rules, such as using a service layer to exhibit business rules in different channels reducing the maintenance cost [32], and others designed to share business rules in

Radio-Frequency Identification (RFID) and enterprise systems [33]. The architecture presented in this paper is distinguished from them by enforcing the same flow of communication regardless of the endpoint, and mainly, by adding the possibility to reuse the entire service and business rules layer in other projects even with a different programming language.

Besides the software architecture it would be possible to reuse business rules through components [34]. This approach was avoided here because they migrate through the software development cycle and have relations with a user interface [35]. On the other hand, there are works that start from the domain and build a software architecture using patterns [36], [37], or through ontologies, dependencies and test tools for building a set of recommended structures to be used in the development or evaluate them [38], [39].

4.1 Business rules management tools

Besides architecture and patterns, business rules deal with, amongst others: (*i*) with a repository that holds, analyzes and reuses process models implemented as a service-oriented architecture [40], (*ii*) custom documentation of generic and reusable design decisions for domain-specific modeling languages [41], and (*iii*) an automatic generation of domain-specific rules from domain models, originating in the need to adapt and promptly implement changes in business [42]. This work path is an important alternative for systems that are not based on specific architecture, or for those that have complemented the structure. Regardless, the use is a relevant technique for the definition of easy access business rules, which would also be useful to link with this work.

4.2 Differences with the domain model pattern

The model layer has similarities with the domain model pattern however the frequent use of this pattern is implemented as a bottom layer (horizontal) and here it is implemented as a side layer (vertical) with no business rules attached. The reason behind this change is centered in the use of the model by others layers, without ignoring or “jumping” layers to obtain access. Although model use is facilitated for other layers it is not allowed to add technological references in the objects, for example Table and Column attributes used in ORM frameworks. In the ORM attributes example if the data access technology is changed the attributes no longer have any meaning in the model, and even if not changed the model is used by other layers; if each one adds content relevant to themselves the objects would be polluted and harder to maintain. In both implemented systems a decoupling form was used to associate objects with tables and columns through the mapper pattern.

4.3 Strength and limitations

This work used the proposed model to develop two systems and achieved the reuse, decoupling and isolation of business rules. The differences between this work and others are the dynamic and decoupled form to reuse business rules through software architecture, and also to incorporate new technologies supporting the evolution of systems. They are dynamic in the sense that the business rules relevant to the systems are implemented as functions or methods in source code, in a specific layer that is irrelevant to the systems and ready to use. In addition to that, the business rules are decoupled because it performs only operations related to the business core, and also, by using the language types (mainly primitive ones) or objects created in the same layer. In the case of complex language types, such as Lists, it is recommended to use those with standard serialization, which tend to mitigate the risk of failure in interoperability operations.

Limitations observed for this work were that technical validation such as length of fields is implemented with the systems business rules. Also, the entire interface was developed using .NET Framework and although they did not invalidate the findings of this work a different line of technology would increase the value of the reuse of the business rules.

4.4 Future work

Future work would be to implement systems using this model with a different language and technologies. Other future work is the evolution of the S1 system used for collecting data through the patient reported outcome instruments, which is already in progress.

5 Conclusion

The designed methodology establishes an approach to organize, reuse and isolate the business rules with a high level of decoupling. Two different systems were developed using the same methodology organized in an architectural system and illustrated how some layers, in a specific level, could be added or removed according to the systems requirements without compromising the results.

References

[1] W. H. Inmon, B. K. O'Neil, and L. Fryman, Business metadata : capturing enterprise knowledge. Elsevier/Morgan Kaufmann, 2008.

[2] R. G. Ross, "IT systems perspective - The business rule approach," Computer (Long. Beach. Calif)., vol. 36, no. 5, pp. 85–87, May 2003.

[3] K. Zhao, S. Ying, L. Zhang, and L. Hu, "Achieving business process and business rules integration using SPL," in 2010 International Conference on Future Information Technology and Management Engineering, 2010, pp. 329–332.

[4] ISO/IEC/IEEE 24765 : 2017(E): ISO/IEC/IEEE International Standard - Systems and software engineering--Vocabulary. IEEE, 2017.

[5] H. S. Hamza and M. E. Fayad, "A novel approach for managing and reusing business rules in business architectures," in The 3rd ACS/IEEE International Conference onComputer Systems and Applications, 2005., 2005, pp. 973–978.

[6] E. Evans, Domain-driven design : tackling complexity in the heart of software. Addison-Wesley, 2004.

[7] M. Fowler, Patterns of enterprise application architecture. Addison-Wesley, 2003.

[8] C.-Z. Lu, G.-S. Zeng, and W.-J. Liu, "Software Evolution Rules with Condition Constrains to Support Component Type Matching Based on Bigraph," Int. J. Softw. Eng. Knowl. Eng., vol. 28, no. 10, pp. 1429–1453, Oct. 2018.

[9] A. Sharma, M. Kumar, and S. Agarwal, “A Complete Survey on Software Architectural Styles and Patterns,” Procedia Comput. Sci., vol. 70, pp. 16–28, Jan. 2015.

[10] A. Sedaghatbaf and M. Abdollahi Azgomi, “Reliability evaluation of UML/DAM software architectures under parameter uncertainty,” IET Softw., vol. 12, no. 3, pp. 236–244, Jun. 2018.

[11] A. MacCormack and D. J. Sturtevant, “Technical debt and system architecture: The impact of coupling on defect-related activity,” J. Syst. Softw., vol. 120, pp. 170–182, Oct. 2016.

[12] S. A. Ebad and M. A. Ahmed, “Measuring stability of object-oriented software architectures,” IET Softw., vol. 9, no. 3, pp. 76–82, Jun. 2015.

[13] M. Galster, A. Eberlein, and M. Moussavi, “Systematic selection of software architecture styles,” IET Softw., vol. 4, no. 5, p. 349, 2010.

[14] Brooks, “No Silver Bullet Essence and Accidents of Software Engineering,” Computer (Long. Beach. Calif)., vol. 20, no. 4, pp. 10–19, Apr. 1987.

[15] D. Garlan, J. M. Barnes, B. Schmerl, and O. Celiku, “Evolution styles: Foundations and tool support for software architecture evolution,” in 2009 Joint Working IEEE/IFIP Conference on Software Architecture & European Conference on Software Architecture, 2009, pp. 131–140.

[16] N. Haile and J. Altmann, “Evaluating investments in portability and interoperability between software service platforms,” Futur. Gener. Comput. Syst., vol. 78, pp. 224–241, Jan. 2018.

[17] C. C. Venters et al., “Software sustainability: Research and practice from a software architecture viewpoint,” J. Syst. Softw., vol. 138, pp. 174–188, Apr. 2018.

[18] L. Bass, P. Clements, and R. Kazman, Software architecture in practice. Addison-Wesley, 2013.

[19] Ali Akbar Serafat and H. Shirazi, “Presenting a meta layer model for separation of crosscutting concerns in aspect oriented architecture,” in 2010 3rd International Conference on Computer Science and Information Technology, 2010, pp. 23–27.

[20] B. Esselink, A Practical guide to localization. John Benjamins Pub. Co, 2000.

[21] Microsoft, “Microsoft technical documentation.” [Online]. Available: https://docs.microsoft.com/en-us.

[22] Oracle, “Java 10 Documentation.” [Online]. Available: https://docs.oracle.com/javase/10/.

[23] F. Rademacher, J. Sorgalla, and S. Sachweh, “Challenges of Domain-Driven Microservice Design: A Model-Driven Perspective,” IEEE Softw., vol. 35, no. 3, pp. 36–43, May 2018.

[24] A. W. Troelsen, COM and .NET interoperability. Apress, 2002.

[25] D. Duggan, Enterprise software architecture and design : entities, services, and resources. Wiley, 2012.

[26] M. Kassab, J. DeFranco, and P. Laplante, “Software Testing Practices in Industry: The State of the Practice,” IEEE Softw., pp. 1–1, 2016.

[27] Zhimin Li, Xiangdong Li, and Minjie Xia, “Design and implementation of data access layer based on design pattern,” in 2010 International Conference on Computer Application and System Modeling (ICCASM 2010), 2010, pp. V14-98-V14-100.

[28] D. Rosenberg and M. Stephens, Use case driven object modeling with UML : theory and practice. Apress, 2007.

[29] StructureMap, “StructureMap documentation.” [Online]. Available: http://structuremap.github.io/documentation/.

[30] S. Ali, B. Soh, and T. Torabi, “Using Software Engineering Principles to Develop Reusable Business Rules,” in 2005 International Conference on Information and Communication Technologies, 2005, pp. 276–283.

[31] W. M. N. Wan-Kadir and P. Loucopoulos, “Relating evolving business rules to software design,” J. Syst. Archit., vol. 50, no. 7, pp. 367–382, Jul. 2004.

[32] Z. Ji, I. Ganchev, and M. O’Droma, “Intelligent Software Architecture for the Service Layer of Wireless Billboard Channels,” in 2009 6th IEEE Consumer Communications and Networking Conference, 2009, pp. 1–2.

[33] X. Zhao, C. Liu, and T. Lin, “Incorporating business logics into RFID-enabled applications,” Inf. Process. Manag., vol. 48, no. 1, pp. 47–62, Jan. 2012.

[34] A. P. Sinha and H. Jain, “Reusing business components and objects for modeling business systems: The influence of decomposition characteristics and analyst experience,” J. Syst. Softw., vol. 131, pp. 550–569, Sep. 2017.

[35] F. Barbier and C. Atkinson, “Business Components,” in Business Component-Based Software Engineering, Boston, MA: Springer US, 2003, pp. 1–26.

[36] D. Lucredio, R. P. M. Fortes, E. S. Almeida, and S. L. Meira, “Designing Domain Architectures for Model-Driven Engineering,” in 2010 Fourth Brazilian Symposium on Software Components, Architectures and Reuse, 2010, pp. 100–109.

[37] M. A. . de Souza and M. A. G. . Ferreira, “Designing reusable rule-based architectures with design patterns,” Expert Syst. Appl., vol. 23, no. 4, pp. 395–403, Nov. 2002.

[38] E. Y. Nakagawa, E. F. Barbosa, and J. C. Maldonado, “Exploring ontologies to support the establishment of reference architectures: An example on software testing,” in 2009 Joint Working IEEE/IFIP Conference on Software Architecture & European Conference on Software Architecture, 2009, pp. 249–252.

[39] T. Lutellier et al., “Measuring the Impact of Code Dependencies on Software Architecture Recovery Techniques,” IEEE Trans. Softw. Eng., vol. 44, no. 2, pp. 159–181, Feb. 2018.

[40] M. La Rosa et al., “APROMORE: An advanced process model repository,” Expert Syst. Appl., vol. 38, no. 6, pp. 7029–7040, Jun. 2011.

[41] B. Hoisl, S. Sobernig, and M. Strembeck, “Reusable and generic design decisions for developing UML-based domain-specific languages,” Inf. Softw. Technol., vol. 92, pp. 49–74, Dec. 2017.

[42] N. Mani, M. Helfert, and C. Pahl, “A Domain-specific Rule Generation Using Model-Driven Architecture in Controlled Variability Model,” Procedia Comput. Sci., vol. 112, pp. 2354–2362, Jan. 2017.

4 ARTIGO DA DISSERTAÇÃO

Research electronic health record databases

Leonardo Pellizzoni[1]; Scheila A. Silva[2]; Asdrubal Falavigna[1]

[1]Health Sciences Postgraduate Program, University of Caxias do Sul (UCS), Caxias do Sul, Rio Grande do Sul, Brazil.

[2]Biotechnology Postgraduate Program, University of Caxias do Sul (UCS), Caxias do Sul, Rio Grande do Sul, Brazil.

ABSTRACT

Introduction: A dataset with patient information allows a comparison between different clinical treatments in many fields of medicine as well as the efficient use of medical resources. Patient-reported outcome measures (PROMs) collect data directly from patients. The data collected through PROMs can be used in clinical practice by helping decision making and tailoring treatments according to the patient's needs.

Objective: To develop electronic health record databases for monitoring clinical or surgical interventions and measuring the quality of life of the patients.

Methodology: Process modeling and specification of system requirements were performed using the Iconix methodology along with the Post-Study System Usability Questionnaire (PSSUQ) to validate the usability and usefulness of the proposed system. The system and the questionnaires were performed in three languages: Brazilian Portuguese, Spanish, and English.

Results: The platform enables the researchers to use the questionnaires defining the time of the data collection according to the needs of each clinical study. The system facilitates the patient answers without any personal interference from smartphones, tablets or computers. The questionnaire' scores were calculated automatically in real time and displayed in graphics on the patients' dashboard.

Conclusion: An electronic health record database enables collecting information on the patient directly from their own devices directly to the database without any interference from researchers and with real-time graphics.

Keywords: Patient-reported outcome measures; health technology assessment; registries; outcomes; quality of life measures

Introduction

Health technology assessment (HTA) collects data focusing on the medical, economic, social and ethical implications of development, diffusion and use of health technology that inform real-world decisions about the value of new technologies, interventions and practices [1]. The patient-reported outcome measures (PROMs) collect data directly from the patients, quantifying the quality of life, perspective on the frequency and severity of their symptoms, and how the disease impacts their functioning [2,3]. The use of clinical registries based on PROMs adds strategic value to medical care by narrowing the gap between the clinician's and patient's view of the clinical reality, and also helps tailor the treatment plans for specific needs [4,5]. The patient needs to perceive the gain to them by answering the questionnaire, otherwise they will not be engaged [6].

There are barriers to prevent the physicians and patients from adopting PROMs assessment, such as lack of time, lack of assistance in data collection, and lack of an electronic database [7]. Electronic databases are developed to overcome those drawbacks, making easier and friendly on both sides, physician and patients, to evaluate and follow the medical care [8,9].

The objective of this paper is to present an innovative electronic health record database with new features, such as automation of the data collection process, enabling physicians to customize the design of questionnaires according to the population, practice types and clinical settings of a new research study, alerting the physician and the patient when it is time for the next data collection, and showing the PROMs results compared to the historical outcomes on a dashboard.

Methods

The methodology was designed to initially define the system language, the software and the architecture of the platform. Later, the authors went through different protocols of clinical study to clearly understand the databank issues and barriers faced during the design and execution of a clinical research project [9]. The innovative changes of the electronic health record database to optimize and facilitate the patient record is described in the following sub-sections.

System development

All the data related to the process were stored in a SQL Server database that was only accessed through a system developed using C# language. The system development used the ICONIX methodology [10] and a layered software architecture focusing on the isolation of business rules and the evolution of the system. The features of the developed system consist in creating questionnaires for later use in the data collection design of the clinical studies with multiple research centers.

Clinical research studies

The clinical study follows the design and the schedule of patient data collection proposed by the researchers. The clinical study has a unique identification code, start date, brief description, research center code, and the research center coordinator. The questionnaires were linked in the study by a plan of application that has a schedule before and after the intervention, specific sequence of questionnaires, time interval for application, and tolerance to send the reminder and alert to the physician and patient.

Questionnaires

The platform was tested with different types of questionnaires. The strategy was to build PROMs with a variable quantity of questions and possible answers. The answer could be single choice, multiple choice, numerical, date and text. The system was designed to not allow advancing to the next question if the answer to that question was not given. The questionnaires were provided in English, Portuguese and Spanish.

The present study used the Oswestry disability index (ODI) and the Euro quality of life 5D (EQ5D) questionnaire [11,12]. Additional information was collected with questions regarding procedure costs, clinical information, surgical and patient’s outcome data.

The ODI scale consists of 10 questions that assess domains of pain with six possible answers whose values range from 0 to 5, with a total score from zero to 100: 0-20 indicates minimal disability; 21-40 indicates moderate disability; 41-60 indicates severe disability; 61-80 indicates crippling back pain; and 81-100 indicates that the patient is either bed-bound or exaggerating their symptoms [11]. The EQ5D is a generic instrument used to measure preference-based health status for health economic analyses in five areas: mobility, self-care, usual activities (work, study, housework, and family or leisure activities), pain/discomfort and anxiety/depression [12].

Patients

The patients registered in the database were linked to the specific clinical study in which they were previously enrolled with an exclusive identification code. The information collected from the patients was sex, date of birth, weight, date of entry into the study, date of the intervention, e-mail, language, and how questionnaires will be collected. The code was automatically suggested by the system which uses the research center code as a prefix followed by the number of patients at that center plus 1.

Authorization and security management

An authorization feature was implemented to determine who can access it and what operations they can perform. The system has a manageable customized control for operations of viewing, saving and deleting information. User’s permissions are directly associated with the responsibilities assumed in the study. Security mechanisms were available in the SQL Server database in which sensitive information was encrypted.

Active follow up

The system supports both paper and electronic data collection. When the collection of data was on paper the questionnaire was printed and handed over to the respondent. Then the answers were recorded in the system by the researchers. For the electronic data collection, the respondent received an email notification containing a weblink that accessed the questionnaire and allowed the answers to be recorded using cellphone, tablet or computer. The notifications were controlled on a daily basis and sent by email to the respondent, investigator and coordinator to avoid the loss of data by not completing the questionnaires within the allotted time.

The patient's data collection schedule is managed automatically by the system using the patient's date of entry into the study, clinical or surgical intervention and the monitoring frequency previously informed in the study design. A virtual schedule was developed to display the questionnaires to be filled as appointments. The data collection period for the questionnaires was elaborated as follows:

$$start = base\ + days\ for\ application$$

$$end = base + days\ for\ application + alert\ days$$

The *base* word in the formula is equivalent to the patient's entry date into the study or their clinical/surgical intervention. This formula is used by the system with information established by the researchers on the clinical study schedule and in the patients' electronic form.

Dashboard

A dashboard was developed to visualize the score of the questionnaires in charts. The dashboard automatically calculated the score for each questionnaire using previously informed for each patient to perform the operation in real time. In each questionnaire the system analyzes the score parameters for each one of the alternatives, their increment values, how to group the questions and the type of calculation. The group types covered in the system are: (i) all questions of the questionnaire, (ii) by questionnaire section or (iii) group of questions chosen manually. The following types were implemented in the calculation of the score: (i) sum up, (ii) sum up and divide by the score and (iii) unite questions score. It is also possible to define legends and percent values for the scores.

Integration with other systems

The answers collected and stored in the database were exported to a CSV file. The data in the CSV file were organized according to the correspondent clinical study, the chronological order of the applied questionnaires, and their questions. The structure of the file has the patient as a line and each one of the questions as a column. The variables that represent the questions have a unique code generated by the system, which is composed by the questionnaire code, the moment when the data were collected, and the number of the question. The mechanism which creates the variable code allows the same questionnaire applied at different times to generate different variable codes.

Applicability and understanding of the use of the system

The developed system was evaluated by the Post-Study System Usability Questionnaire (PSSUQ) which measures the user's usability and satisfaction in relation to the system with 96% reliability [13]. The PSSUQ questionnaire has 19 questions and uses a Likert scale from 1 (strongly disagree) to 7 (strongly agree). Higher scores indicate better ratings in all the scores.

The PSSUQ questionnaire produces four scores through the average of the responses (*i*) overall satisfaction score (Overall): average of all the 19 questions, (*ii*) system usefulness (SysUse): average of the responses to question 1 through 8, (*iii*) information quality (InfoQual): average of the responses to question 9 through 15, and (*iv*) interface quality (InterQual): average of the responses to question 16 through 18 [13].

PSSUQ was validated to Portuguese from Portugal in 2015 [14]. PSSUQ was translated from English into Portuguese and Spanish by two independent translators in each language. The translations were revised and compared by a third translator both in Portuguese and Spanish and when there were differences in the text, there was an agreement as to the best final version. A translator with English as native language and without knowledge of the original version of PSSUQ performed the back translation from Portuguese and Spanish to English. A committee formed by two researchers assessed the back translation and compared it with the original version of PSSUQ and, if necessary, adjustments were made in the final version of PSSUQ in Portuguese and Spanish. The final version in Brazilian Portuguese was compared with the validated version of Portugal Portuguese.

A minimum of five people are necessary to validate a system using PSSUQ [15,16]. PSSUQ was answered by eight people of different ages and applied at two different moments after inclusion into the platform of ten and twenty cases.

Results

The management of clinical studies and monitoring process is automated in a system which is accessed through a website (Figure 1). Once the researcher has logged into the website a schedule is displayed on the home page and also a toolbar with another system's features, according to the level of responsibility assigned to that researcher. All the pending questionnaires related to the researcher's clinical study are grouped by patient and shown in the schedule (Figure 1). A click on the appointment enables the researcher to have a detailed view of the questionnaires and deadline.

The system structure supports multiple clinical studies simultaneously and, for each study, multiple research centers. The data collection process for several studies with different questionnaires could be managed through this platform without crunching the data. The website was self-adaptable according to the language of the researcher in English, Portuguese or Spanish.

Clinical studies

The organization of clinical studies in the system was shown according to the descriptive information of the study, the research center that coordinates the study, the enrolled patients, and the participating research centers. The studies recorded on the platform remain available on a list according to researcher permissions. All the participating research centers linked in the study work together with the coordinator center.

The research centers have an exclusive code used for identifying the center that allocates those patients. The person who will be in charge of the research and inclusion of the data was allowed access to the platform. Each center can have one coordinator and one principal investigator. The security permissions and management of patients in the process were controlled automatically by the system, since each of the patients was restricted to a single research center.

Another key part of the clinical study is the selection of the PROMs from the previously registered questionnaires which will be used in that study to collect data. The application of the questionnaire is designed according to the researchers' preferences. The schedule (before and after the intervention), order of the questionnaires, application period, and the alert days (Figure 2) were defined. A data collection schedule was created automatically for each patient enrolled in the study. The pre or postintervention schedule allows the system to know which date, entry or intervention, has to be used as a starting point to create the schedule according to the days for application.

Schedules displayed in Figure 3 is a result of using the study design shown in Figure 2. The result in Figure 3 was related to a Patient A with both entry and intervention date on 2/3/2019.

Questionnaires

The questionnaire feature was flexibility, and it was possible to create and record several measuring instruments in different health domains. Once the questionnaires were informed in the system, they could be used as many times as necessary and reused in other studies. The researchers could modify the questionnaire structure (i.e. questions, alternatives, etc.) until no data were collected. Questionnaire modifications were blocked by the system once the data collection was performed, to avoid different answers. All the questionnaires recorded in the system by the researchers were displayed on a list.

Some questions from the ODI questionnaire are shown in Figure 4 in English, Portuguese and Spanish. Regardless of the questionnaire each question has the option of recording your information in 3 languages, to be shown at a later moment of data collection automatically according to the respondent's language.

Patients

The patients are shown to the researcher according to their permissions that were defined at the beginning of the study. Once the patient has been registered in the study they cannot be removed, they can only have their situation modified. Besides the control for patient exclusion, it was not allowed to change the identification code, the clinical study, and the research center responsible for that patient. In the patient's electronic form were displayed all the clinical visits and questionnaire schedules created by the system.

Authorization and security management

The researchers receive one user and password that are associated with an authorization profile. Researchers who coordinate studies had the user created manually and, for those who work in a participating research center, the user was created by an automatic form. Automatic creation of users is done when a new research center has been registered in the system with an e-mail that was not recognized in the database. The researcher receives an e-mail which is sent automatically and contains information to access the system. According to the permissions, the systems control the access among the researchers and allow specific operations to be performed.

Active follow up

Every day an automated process was executed to notify by email patients who have pending questionnaires to answer. The email is sent in the patients's language and has only one link which allows access to all questionnaires pending in that period.

It is not necessary to provide user and password to the patient answering the questions because the link has a security key embedded. Each of the notifications sent has a link with a different security key, even if it is for the same patient. Regardless of the number of questionnaires to be answered in that period the system sent only one e-mail. Consequently, when the answers for one questionnaire are saved in the database the system automatically starts the next questionnaire if there is one.

The patient answers the questions designed by the researchers in the questionnaires in his own language. One question is displayed to the patient at a time according to the example in Figure 5.

Even with the e-mail notifications to the patients the system sends an e-mail to the coordinator and investigator for notification. A message of alert is automatically activated and sent to the coordinator, principal investigator, and researchers responsible for the patient when there are 5 days left to the deadline and the patient has not answered the questionnaires.

The questionnaire's access link becomes invalid the moment the data collection has been concluded or when it was accessed in a period different from that designated to collect the data. Both cases display messages informing about the situation. In cases when the data are collected by printed questionnaires the researchers record the information in the database manually.

Dashboard

The entire design of the questionnaire to collect data in the study was displayed on the patient's dashboard. Besides the questions informed in the questionnaire the researchers provide parameters to allow the system to calculate the questionnaire's score. In addition to the score parameters researchers can inform chart legend and specify the custom group of questions to be displayed in the chart. Using the informed parameters, the system calculates the questionnaire scores in real time and displays them in charts as shown in Figure 6. The X axis represents the moment of assessment. The chart can be visualized directly in the system or it is possible to print or export it to PDF.

Integration with other systems

Data export was performed through three steps and executed as many times as necessary. The first step was the selection of the clinical study to export the patients' answers. The wizard displays only those studies where the researcher is the principal coordinator. The second step was the selection of the patients' status to export. The researcher can filter the following status: participating, dropout, study related death, and death for another reason. The last step was the selection of the language for the legend of the variables. The data were organized according to the plan of measuring instruments. The system generates all the variables and places a blank value in unanswered questionnaires, and later when they have been answered and exported again the value informed by the patients will substitute the blank value. It was not necessary to have the previous CSV file to export again. The CSV file also has a list with the meanings of each coding that identifies the variables with the name, questionnaire and the moment of collection.

Figure 7 shows an example of the structure and data of the exported file. The first three variables were related to patient identification. After the identification there were variables that correspond to each one of the questionnaires' questions according to their chronological order of collection. The legend of the variables in the language selected in the wizard is displayed at the end of the file.

Applicability and understanding the use of the system

A total of eight users from the medical field evaluated the system, four physicians and four medical students. The mean score for the general use of the system was very well evaluated with the value of 6.1 and with a standard deviation (SD) of 1.1. The mean subscales scores detailed per group and the standard deviation are described in Table 1. There was no difference between the first and second assessment, showing that the learning curve with the system is close to ten patients.

Discussion

The use of a custom-made electronic database for data collection is addressed by works on different health domains [17–19] and essentially even if partially, they have common features among them. The present digital platform is distinguished by promoting a flexible and automated process that actively collects data for multiple clinical studies for different health domains using a single system. The measuring instruments may vary according to the purpose of each research, although the researchers and physicians could freely and directly manage the questionnaires [20,21].

The RedCap and the EpiData also allows a flexible use of questionnaires in data collection [22,23]. This work differs from RedCap because it was multilingual and allows informing the questionnaire in English, Portuguese and Spanish, according to the respondent's language. In addition, the system uses the design of the study informed by the researchers to automatically schedule and notify each of the patients who was enrolled in that study. As mentioned EpiData allows custom questionnaires, however its process focuses on the digitalization of the data that were previously collected on paper questionnaires [23]. This work differs from EpiData by allowing the collection of data directly by patients, physicians and researchers who can use the system at the same time.

The collaborative activity performed using the network to exchange information among research centers is timely, useful, and makes it possible to obtain the crucial multicenter information that helps the decision-making, such as the National Neurosurgery Quality and Outcomes (N²QOD) group and the ImproveCareNow [24,25]. The system developed for the Evaluation of Integrated Cardiac Imaging (EVINCI) study organized the researchers and multiple research centers, with different localization and responsibilities, to work together with a database [26]. The organization of the research centers in this work enabled collaborative work among multiple health professionals according to custom responsibilities designed for each study.

Besides the data collection, the system provides a dashboard, reminders for completing the questionnaire, sharing and receiving automatically database information, and integrating the data with statistical analysis software. The system did not perform the statistical analysis, however it exports the collected data in CSV format, which allows it to integrate with software like SPSS and Excel or use the data in other programming languages.

Conclusion

The present system allows a friendly and flexible use of PROMs according to the population, needs in practice and clinical settings. The platform promotes active and direct data collection from patients and physicians in English, Portuguese and Spanish. The questionnaires used in the study were created and maintained by physicians. The time elapsed for data collection was defined according to the study design. The dashboard displays the evolution of the outcome in real time by calculating the score of each questionnaire automatically. Due to the flexible nature of the system and its process we believe that other health domains besides spine care, could use the same system to collect data.

Acknowledgements

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

References

1. Drummond MF, Schwartz JS, Jönsson B, Luce BR, Neumann PJ, Siebert U, et al. Key principles for the improved conduct of health technology assessments for resource allocation decisions. Int J Technol Assess Health Care. 2008 Jul 4;24(03):244–58.

2. Spertus J. Barriers to the Use of Patient-Reported Outcomes in Clinical Care. Circ Cardiovasc Qual Outcomes. 2014 Jan 1;7(1):2–4.

3. Falavigna A, Teles AR, Braga GL de, Barazzetti DO, Lazzaretti L, Tregnago AC. Instrumentos de avaliação clínica e funcional em cirurgia da coluna vertebral. Coluna/Columna. 2011;10(1):62–7.

4. Falavigna A, Dozza DC, Teles AR, Wong CC, Barbagallo G, Brodke D, et al. Current Status of Worldwide Use of Patient-Reported Outcome Measures (PROMs) in Spine Care. World Neurosurg. 2017 Dec;108:328–35.

5. Falavigna A, Righesso Neto O, Teles AR. Avaliação clínica e funcional no pré-operatório de doenças degenerativas da coluna vertebral. Coluna/Columna. 2009 Sep;8(3):245–53.

6. Santana MJ, Haverman L, Absolom K, Takeuchi E, Feeny D, Grootenhuis M, et al. Training clinicians in how to use patient-reported outcome measures in routine clinical practice. Qual Life Res. 2015 Jul 15;24(7):1707–18.

7. Teles AR, Righesso O, Gullo MCR, Ghogawala Z, Falavigna A. Perspective of Value-Based Management of Spinal Disorders in Brazil. World Neurosurg. 2016 Mar 1;87:346–54.

8. Goeree R, He J, O'Reilly D, Tarride J-E, Xie F, Lim M, et al. Transferability of health technology assessments and economic evaluations: a systematic review of approaches for assessment and application. Clinicoecon Outcomes Res. 2011;3:89–104.

9. Teles AR, Khoshhal KI, Falavigna A. Why and how should we measure outcomes in spine surgery? J Taibah Univ Med Sci. 2016 Apr 1;11(2):91–7.

10. Rosenberg D, Stephens M. Use case driven object modeling with UML : theory and practice. Apress; 2007. 438 p.

11. Fairbank JC, Pynsent PB. The Oswestry Disability Index. Spine (Phila Pa 1976). 2000 Nov 15;25(22):2940–52; discussion 2952.

12. Coretti S, Ruggeri M, McNamee P. The minimum clinically important difference for EQ-5D index: a critical review. Expert Rev Pharmacoecon Outcomes Res. 2014 Apr 13;14(2):221–33.

13. Lewis JR. Psychometric Evaluation of the PSSUQ Using Data from Five Years of Usability Studies. Int J Hum Comput Interact. 2002 Sep;14(3–4):463–88.

14. Rosa AF, Martins AI, Costa V, Queiros A, Silva A, Rocha NP. European Portuguese validation of the Post-Study System Usability Questionnaire (PSSUQ). In: 2015 10th Iberian Conference on Information Systems and Technologies (CISTI). IEEE; 2015. p. 1–5.

15. Lewis JR, R. J. Sample sizes for usability tests. interactions. 2006 Nov 1;13(6):29.

16. Turner CW, Lewis JR, Nielsen J. Determining Usability Test Sample Size. Int Encycl Ergon Hum Factors. 2006;3:3084–8.

17. Azad TD, Kalani M, Wolf T, Kearney A, Lee Y, Flannery L, et al. Building an electronic health record integrated quality of life outcomes registry for spine surgery. J Neurosurg Spine. 2016 Jan;24(1):176–85.

18. Winget M, Kincaid H, Lin P, Li L, Kelly S, Thornquist M. A web-based system for managing and co-ordinating multiple multisite studies. Clin Trials J Soc Clin Trials. 2005 Feb 3;2(1):42–9.

19. Nesbitt G, McKenna K, Mays V, Carpenter A, Miller K, Williams M, et al. The Epilepsy Phenome/Genome Project (EPGP) informatics platform. Int J Med Inform. 2013 Apr;82(4):248–59.

20. Godil SS, Parker SL, Zuckerman SL, Mendenhall SK, McGirt MJ. Accurately measuring the quality and effectiveness of cervical spine surgery in registry efforts: determining the most valid and responsive instruments. Spine J. 2015 Jun 1;15(6):1203–9.

21. Prinsen CAC, Vohra S, Rose MR, Boers M, Tugwell P, Clarke M, et al. How to select outcome measurement instruments for outcomes included in a “Core Outcome Set” – a practical guideline. Trials. 2016 Dec 13;17(1):449.

22. Harris PA, Taylor R, Thielke R, Payne J, Gonzalez N, Conde JG. Research electronic data capture (REDCap)--a metadata-driven methodology and workflow process for providing translational research informatics support. J Biomed Inform. 2009 Apr;42(2):377–81.

23. Singh S. Review of epidata entry and analysis freewares. Indian J Community Med. 2009 Jan;34(1):76–7.

24. Asher AL, Speroff T, Dittus RS, Parker SL, Davies JM, Selden N, et al. The National Neurosurgery Quality and Outcomes Database (N2QOD). Spine (Phila Pa 1976). 2014 Oct 15;39(22 Suppl 1):S106–16.

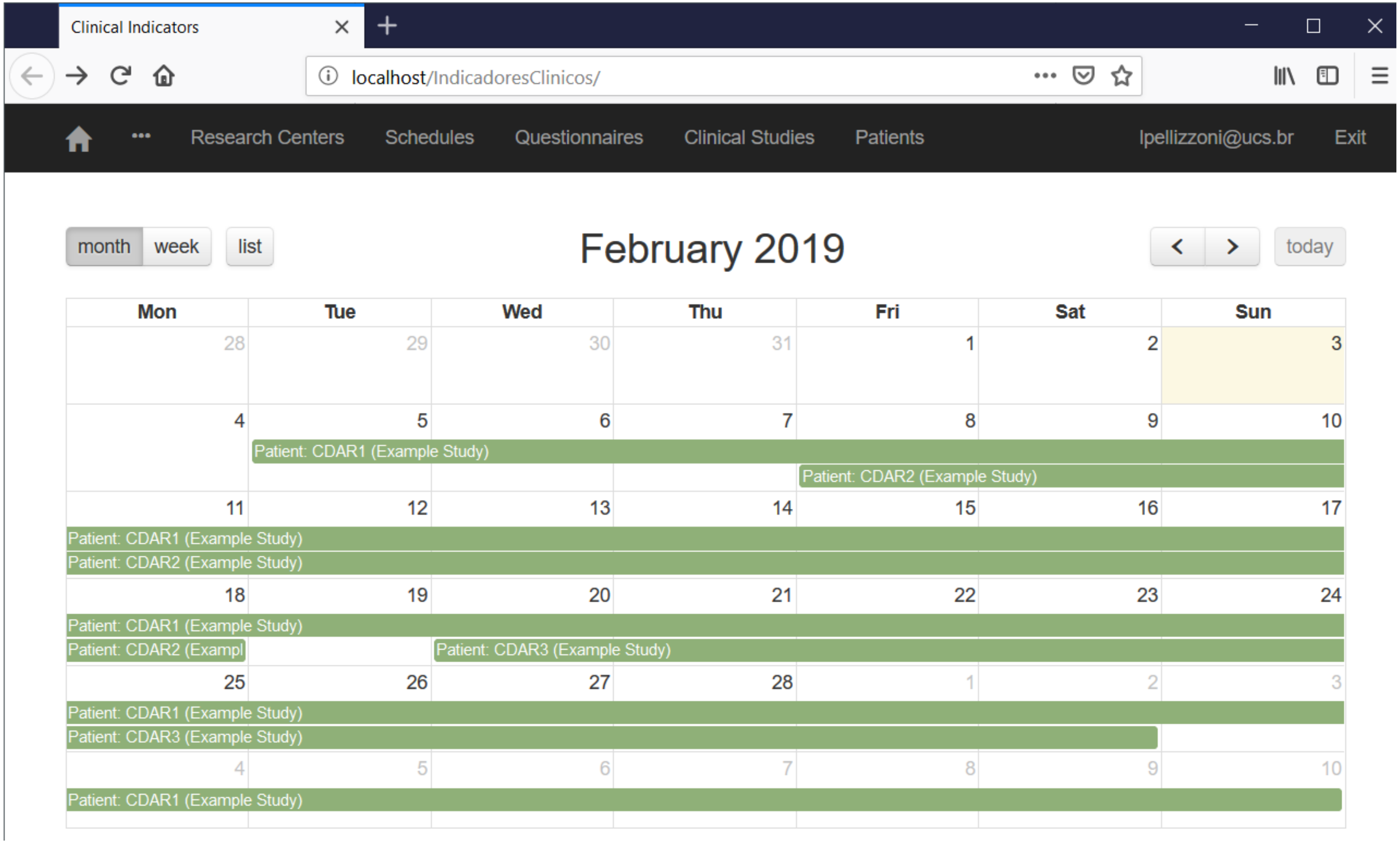
25. Crandall W, Kappelman MD, Colletti RB, Leibowitz I, Grunow JE, Ali S, et al. ImproveCareNow: The development of a pediatric inflammatory bowel disease improvement network. Inflamm Bowel Dis. 2011 Jan;17(1):450–7.

26. Marinelli M, Positano V, Lorenzoni V, Caselli C, Mangione M, Marcheschi P, et al. A modular informatics platform for effective support of collaborative and multicenter studies in cardiology. Health Informatics J. 2016 Dec 26;22(4):1083–100.

Table 1. System evaluation using the Post-Study System Usability Questionnaire scores.

		Mean	SD
General	Overall satisfaction	6,1	1.1
	System usefulness	6,2	1.1
	Information quality	5,9	1.1
	Interface quality	6	1.1
Physicians	Overall satisfaction	6,3	0,4
	System usefulness	6,6	0,3
	Information quality	6,1	0,7
	Interface quality	6,3	0,4
Medical students	Overall satisfaction	5,8	1,5
	System usefulness	5,8	1,5
	Information quality	5,8	1,5
	Interface quality	5,7	1,5

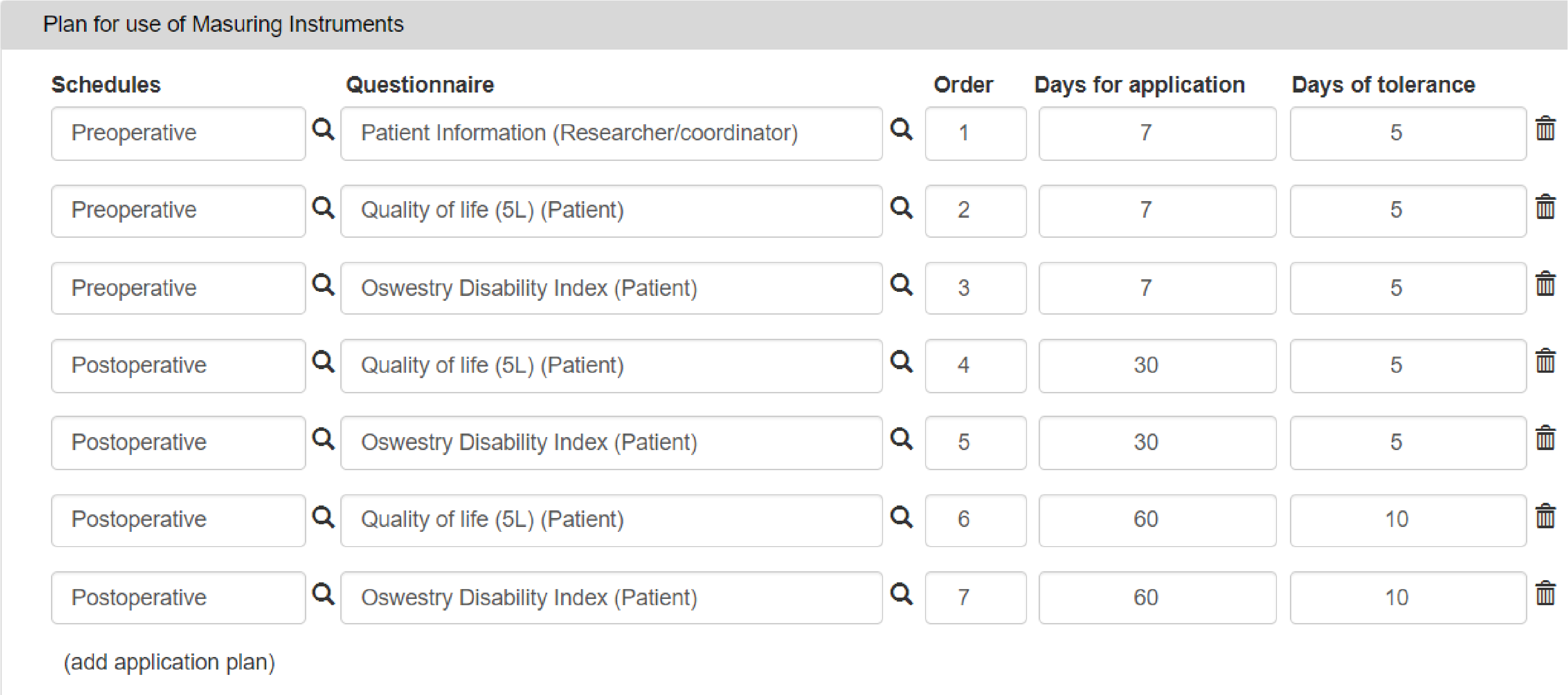
Figure 1. System homepage view.



February 2019

Mon	Tue	Wed	Thu	Fri	Sat	Sun
28	29	30	31	1	2	3
4	5 Patient: CDAR1 (Example Study)	6	7	8 Patient: CDAR2 (Example Study)	9	10
11 Patient: CDAR1 (Example Study) Patient: CDAR2 (Example Study)	12	13	14	15	16	17
18 Patient: CDAR1 (Example Study) Patient: CDAR2 (Exampl	19	20 Patient: CDAR3 (Example Study)	21	22	23	24
25 Patient: CDAR1 (Example Study) Patient: CDAR3 (Example Study)	26	27	28	1	2	3
4 Patient: CDAR1 (Example Study)	5	6	7	8	9	10

Figure 2. Example of the questionnaires that compose the design of the study.

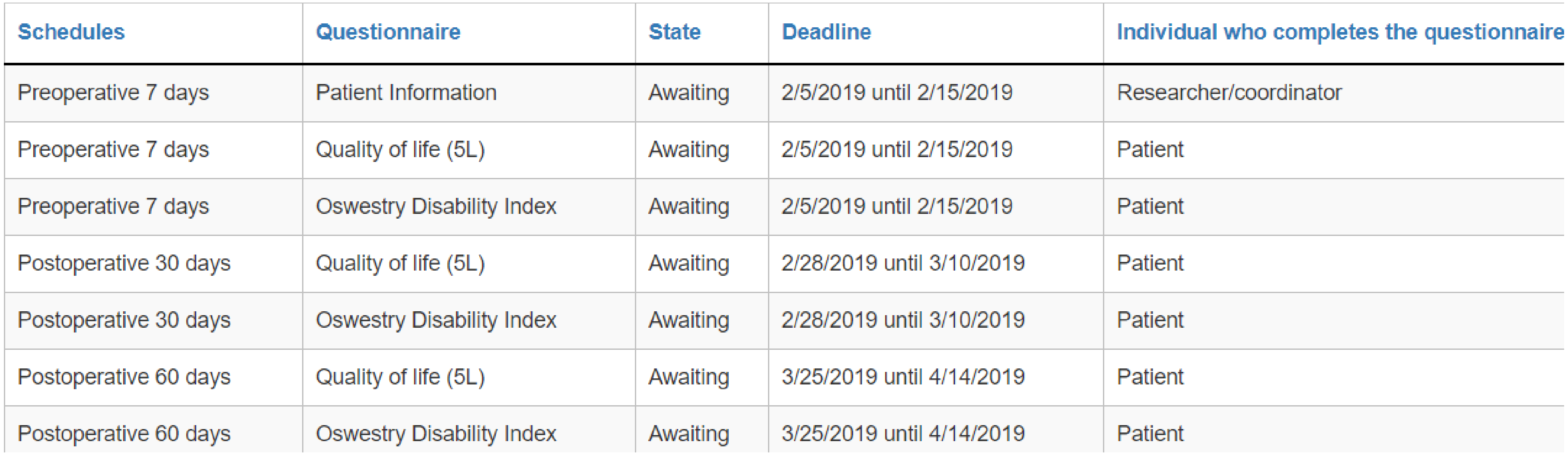


Plan for use of Masuring Instruments

Schedules		Questionnaire		Order	Days for application	Days of tolerance	
Preoperative	🔍	Patient Information (Researcher/coordinator)	🔍	1	7	5	🗑
Preoperative	🔍	Quality of life (5L) (Patient)	🔍	2	7	5	🗑
Preoperative	🔍	Oswestry Disability Index (Patient)	🔍	3	7	5	🗑
Postoperative	🔍	Quality of life (5L) (Patient)	🔍	4	30	5	🗑
Postoperative	🔍	Oswestry Disability Index (Patient)	🔍	5	30	5	🗑
Postoperative	🔍	Quality of life (5L) (Patient)	🔍	6	60	10	🗑
Postoperative	🔍	Oswestry Disability Index (Patient)	🔍	7	60	10	🗑

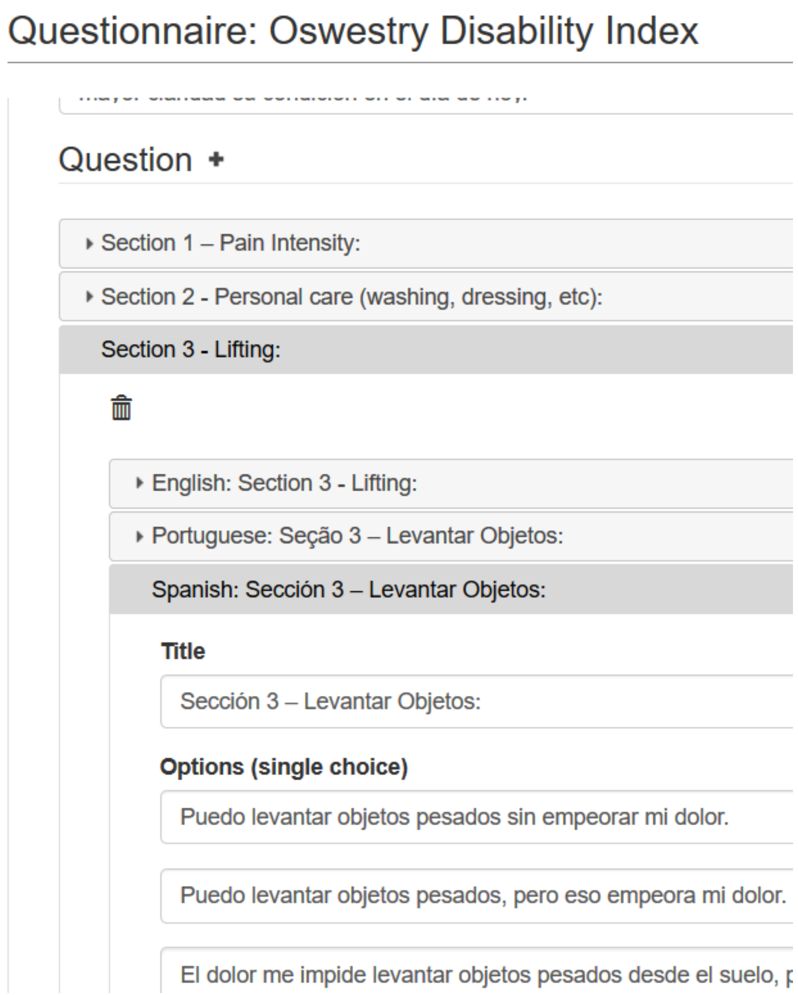
(add application plan)

Figure 3. Schedules created by the system for patient A.



Schedules	Questionnaire	State	Deadline	Individual who completes the questionnaire
Preoperative 7 days	Patient Information	Awaiting	2/5/2019 until 2/15/2019	Researcher/coordinator
Preoperative 7 days	Quality of life (5L)	Awaiting	2/5/2019 until 2/15/2019	Patient
Preoperative 7 days	Oswestry Disability Index	Awaiting	2/5/2019 until 2/15/2019	Patient
Postoperative 30 days	Quality of life (5L)	Awaiting	2/28/2019 until 3/10/2019	Patient
Postoperative 30 days	Oswestry Disability Index	Awaiting	2/28/2019 until 3/10/2019	Patient
Postoperative 60 days	Quality of life (5L)	Awaiting	3/25/2019 until 4/14/2019	Patient
Postoperative 60 days	Oswestry Disability Index	Awaiting	3/25/2019 until 4/14/2019	Patient

Figure 4. Example of the questionnaire template registered in 3 languages.



Questionnaire: Oswestry Disability Index

mayor claridad su condición en el día de hoy.

Question ✚

- Section 1 – Pain Intensity:
- Section 2 - Personal care (washing, dressing, etc):
- Section 3 - Lifting:

🗑

- English: Section 3 - Lifting:
- Portuguese: Seção 3 – Levantar Objetos:
- Spanish: Sección 3 – Levantar Objetos:

Title

Sección 3 – Levantar Objetos:

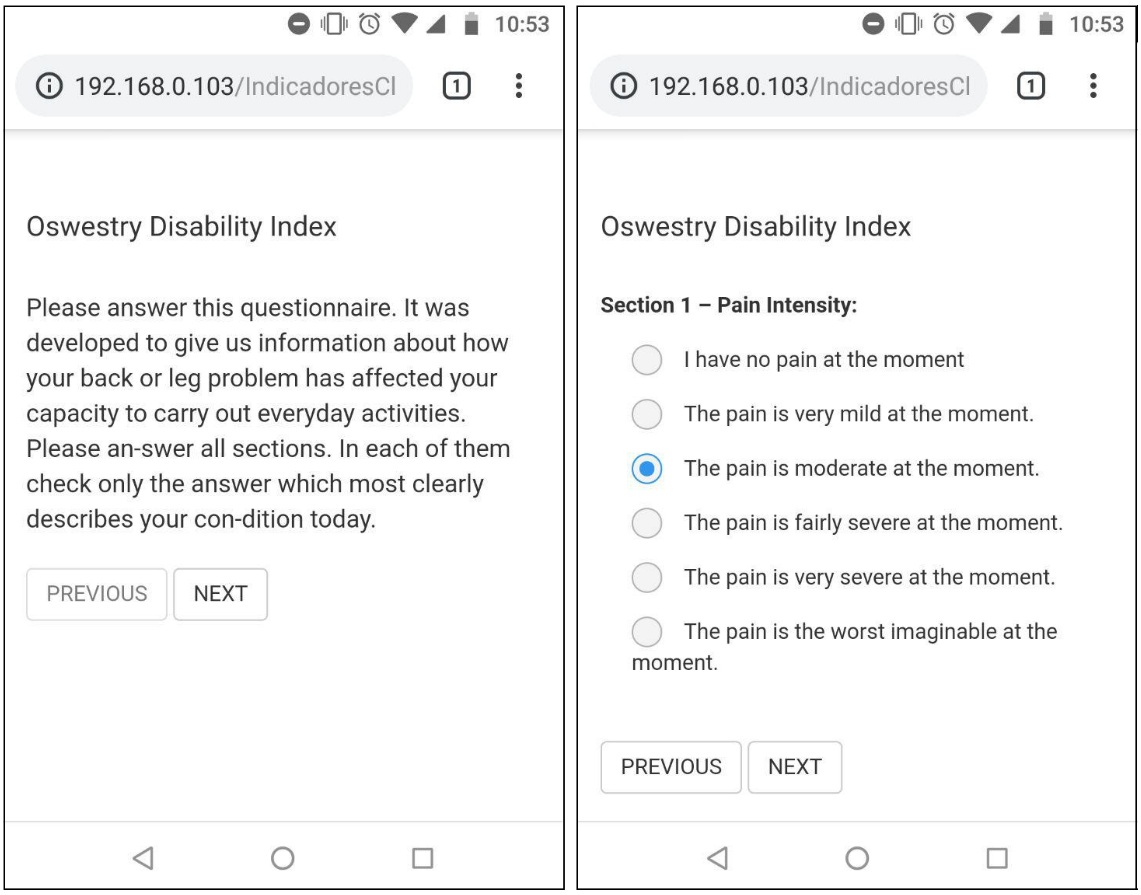
Options (single choice)

Puedo levantar objetos pesados sin empeorar mi dolor.

Puedo levantar objetos pesados, pero eso empeora mi dolor.

El dolor me impide levantar objetos pesados desde el suelo, p

Figure 5. Access to the ODI questionnaire in the mobile browser performed by the patient.



Oswestry Disability Index

Please answer this questionnaire. It was developed to give us information about how your back or leg problem has affected your capacity to carry out everyday activities. Please an-swer all sections. In each of them check only the answer which most clearly describes your con-dition today.

Oswestry Disability Index

Section 1 – Pain Intensity:

○ I have no pain at the moment

○ The pain is very mild at the moment.

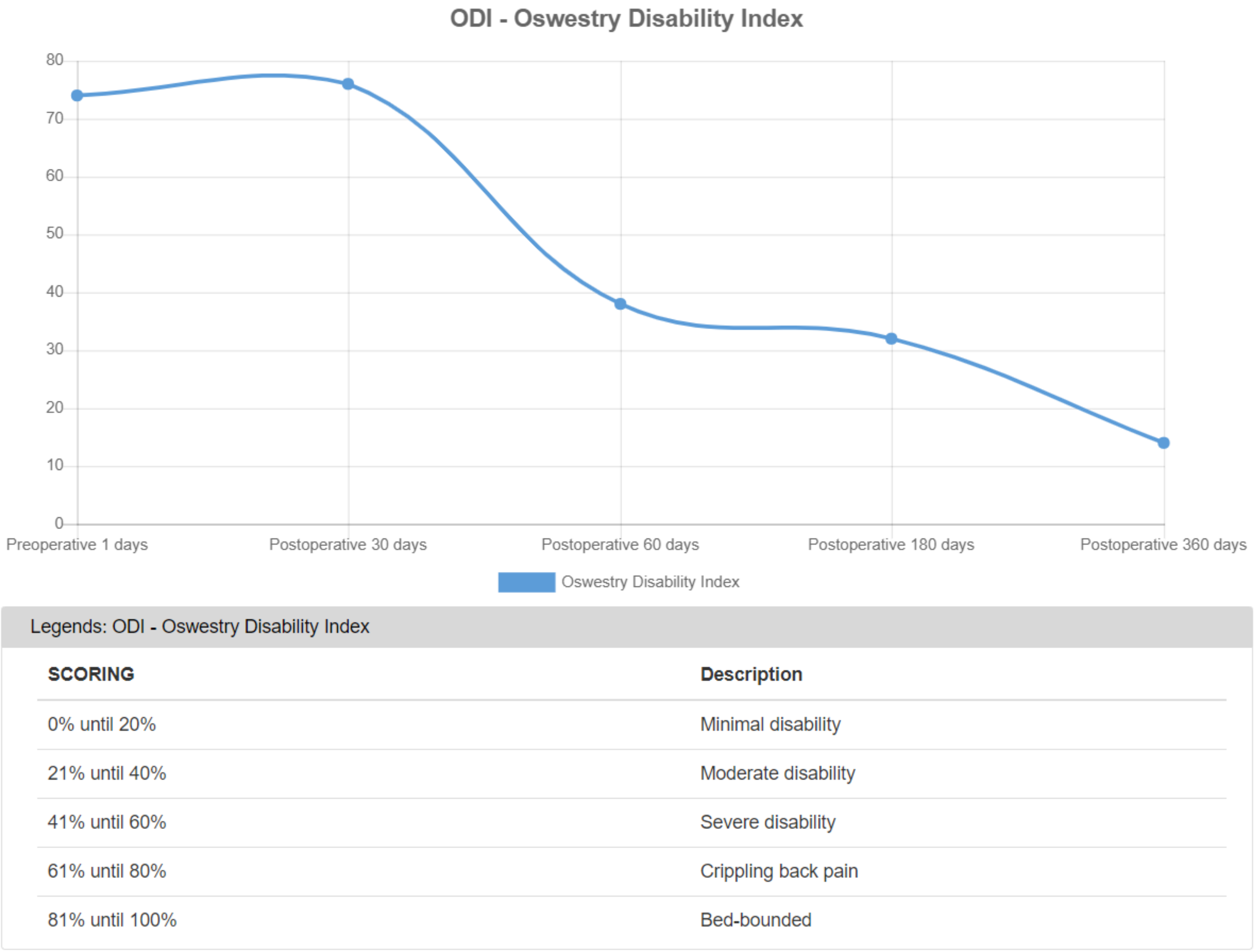
● The pain is moderate at the moment.

○ The pain is fairly severe at the moment.

○ The pain is very severe at the moment.

○ The pain is the worst imaginable at the moment.

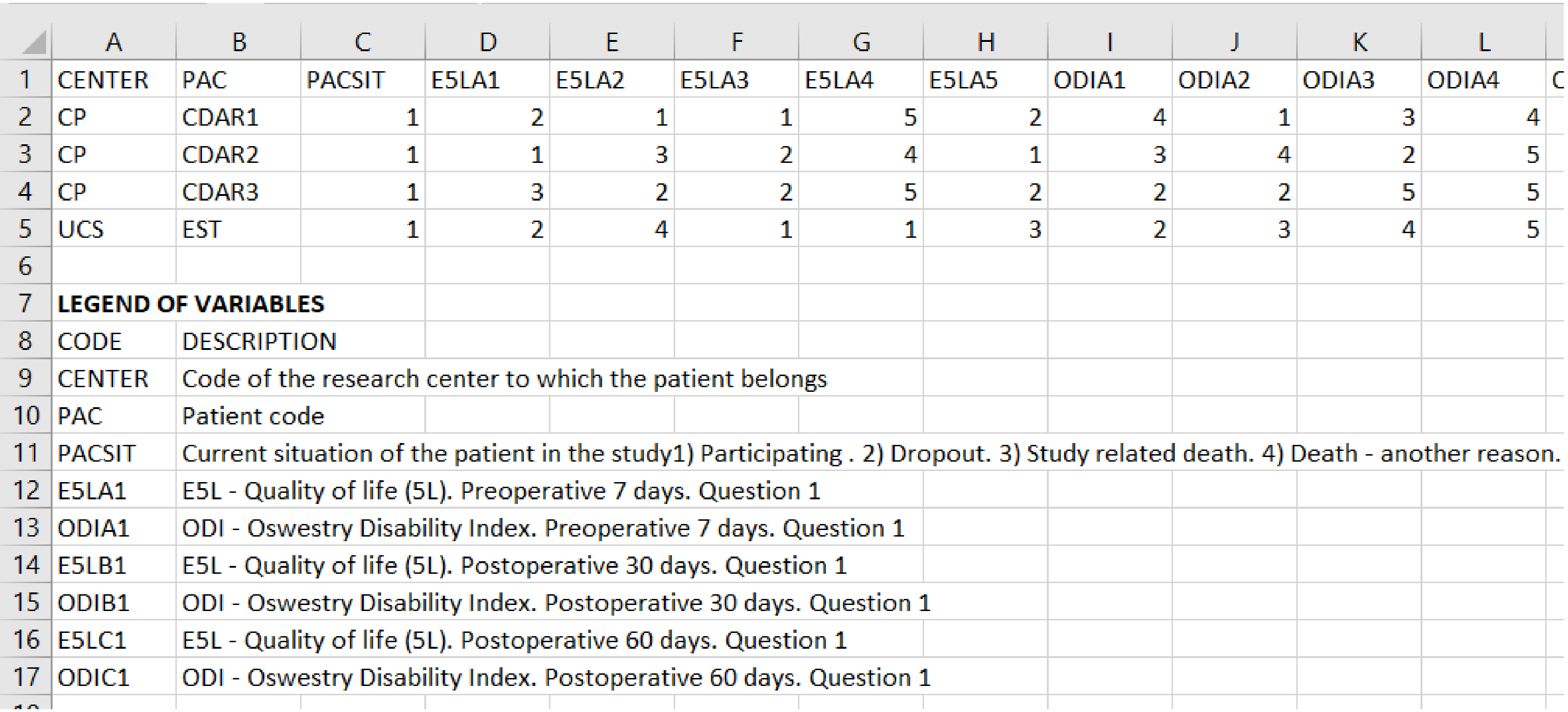
Figure 6. Chart of the ODI score generated by the system for one patient in his dashboard.



Legends: ODI - Oswestry Disability Index

SCORING	Description
0% until 20%	Minimal disability
21% until 40%	Moderate disability
41% until 60%	Severe disability
61% until 80%	Crippling back pain
81% until 100%	Bed-bounded

Figure 7. Example of a CSV file exported by the system.



	A	B	C	D	E	F	G	H	I	J	K	L	
1	CENTER	PAC	PACSIT	E5LA1	E5LA2	E5LA3	E5LA4	E5LA5	ODIA1	ODIA2	ODIA3	ODIA4	C
2	CP	CDAR1	1	2	1	1	5	2	4	1	3	4	
3	CP	CDAR2	1	1	3	2	4	1	3	4	2	5	
4	CP	CDAR3	1	3	2	2	5	2	2	2	5	5	
5	UCS	EST	1	2	4	1	1	3	2	3	4	5	
6													
7	**LEGEND OF VARIABLES**												
8	CODE	DESCRIPTION											
9	CENTER	Code of the research center to which the patient belongs											
10	PAC	Patient code											
11	PACSIT	Current situation of the patient in the study1) Participating . 2) Dropout. 3) Study related death. 4) Death - another reason.											
12	E5LA1	E5L - Quality of life (5L). Preoperative 7 days. Question 1											
13	ODIA1	ODI - Oswestry Disability Index. Preoperative 7 days. Question 1											
14	E5LB1	E5L - Quality of life (5L). Postoperative 30 days. Question 1											
15	ODIB1	ODI - Oswestry Disability Index. Postoperative 30 days. Question 1											
16	E5LC1	E5L - Quality of life (5L). Postoperative 60 days. Question 1											
17	ODIC1	ODI - Oswestry Disability Index. Postoperative 60 days. Question 1											

5 CONSIDERAÇÕES FINAIS E PERSPECTIVAS FUTURAS

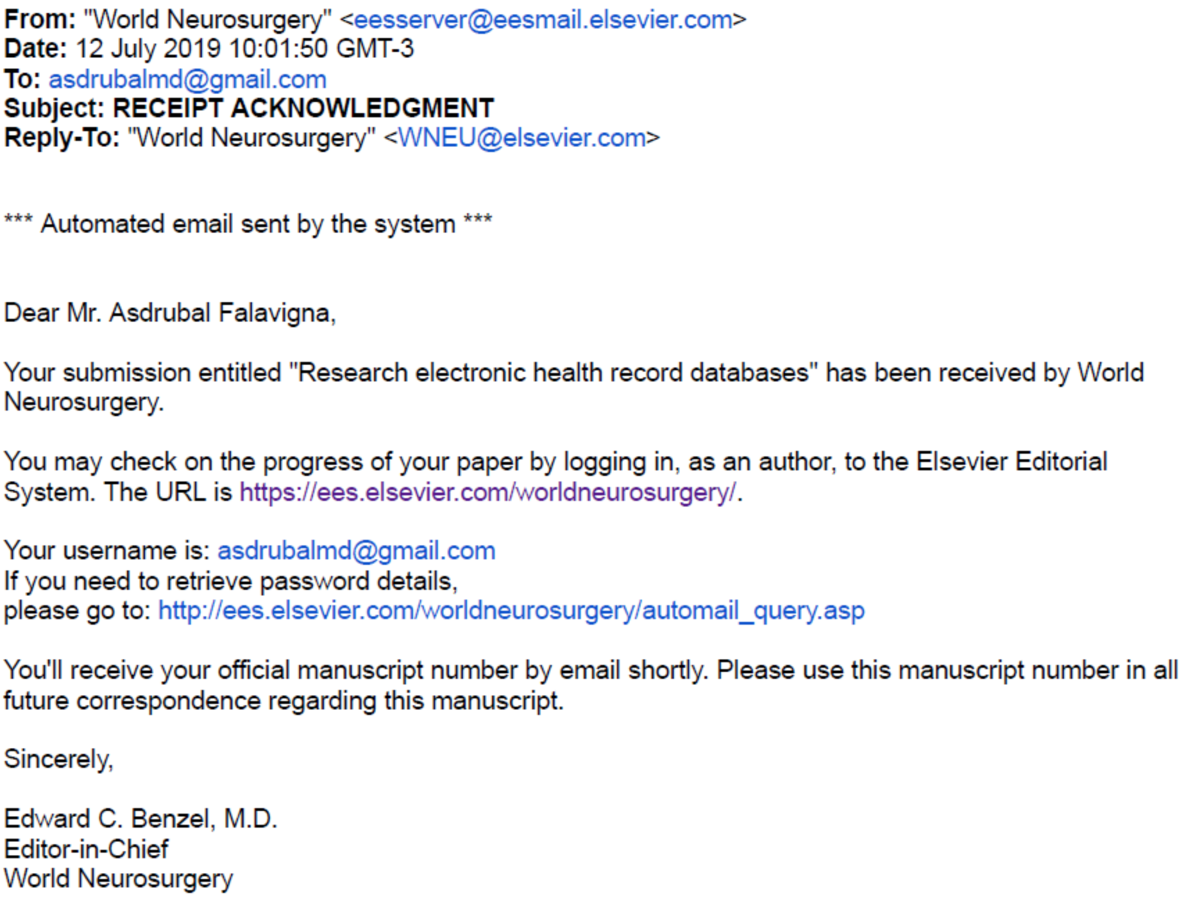
O processo criado e implementado através do sistema auxilia a coleta ativa e gerenciamento de informações de forma automatizada, armazenando-as em estruturas de dados apropriadas e interagindo com pacientes e pesquisadores em 3 idiomas. Este trabalho permite a utilização de questionários de diferentes domínios da saúde de forma flexível, isto é, inseridos e alterados diretamente no sistema, além de permitir um cronograma para coleta de acordo com o delineamento de cada estudo clínico. Um feedback em tempo real foi proporcionado através do cálculo automático de pontuações dos questionários coletados para exibição em gráficos de acompanhamento.

Perspectivas futuras:

- Agregar algoritmos de inteligência artificial diretamente no sistema desenvolvido;
- Usar o sistema em outros em outros estudos clínicos que utilizem diferentes tipos de questionários;
- Utilizar o sistema como ferramenta para coleta de dados em um estudo clínico prospectivo.

ANEXOS

Nesta seção se encontram os comprovantes de submissão. Um referente ao artigo da dissertação submetido a revista **World Neurosurgery (ISSN: 1878-8750)** e outro referente ao artigo sobre a arquitetura de software submetido na revista **International Journal of Software Engineering and Knowledge Engineering (ISSN: 1793-6403)**.



From: "World Neurosurgery" <eesserver@eesmail.elsevier.com>
Date: 12 July 2019 10:01:50 GMT-3
To: asdrubalmd@gmail.com
Subject: RECEIPT ACKNOWLEDGMENT
Reply-To: "World Neurosurgery" <WNEU@elsevier.com>

*** Automated email sent by the system ***

Dear Mr. Asdrubal Falavigna,

Your submission entitled "Research electronic health record databases" has been received by World Neurosurgery.

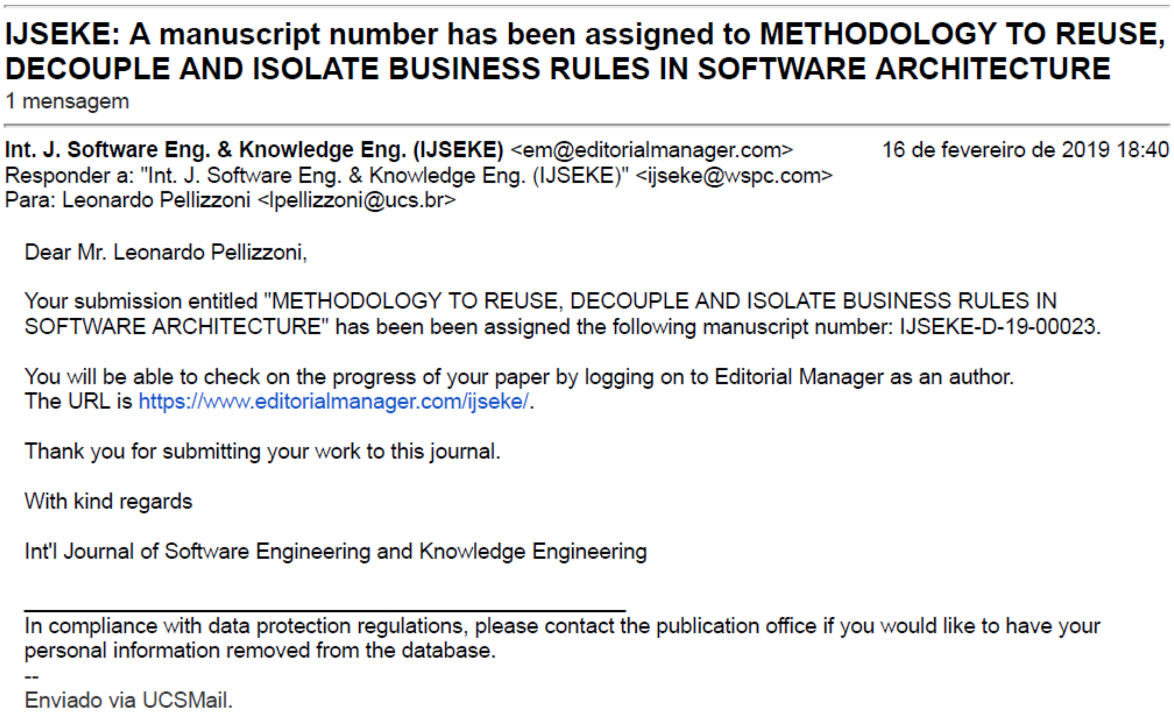
You may check on the progress of your paper by logging in, as an author, to the Elsevier Editorial System. The URL is https://ees.elsevier.com/worldneurosurgery/.

Your username is: asdrubalmd@gmail.com
If you need to retrieve password details,
please go to: http://ees.elsevier.com/worldneurosurgery/automail_query.asp

You'll receive your official manuscript number by email shortly. Please use this manuscript number in all future correspondence regarding this manuscript.

Sincerely,

Edward C. Benzel, M.D.
Editor-in-Chief
World Neurosurgery



IJSEKE: A manuscript number has been assigned to METHODOLOGY TO REUSE, DECOUPLE AND ISOLATE BUSINESS RULES IN SOFTWARE ARCHITECTURE
1 mensagem

Int. J. Software Eng. & Knowledge Eng. (IJSEKE) <em@editorialmanager.com> 16 de fevereiro de 2019 18:40
Responder a: "Int. J. Software Eng. & Knowledge Eng. (IJSEKE)" <ijseke@wspc.com>
Para: Leonardo Pellizzoni <lpellizzoni@ucs.br>

Dear Mr. Leonardo Pellizzoni,

Your submission entitled "METHODOLOGY TO REUSE, DECOUPLE AND ISOLATE BUSINESS RULES IN SOFTWARE ARCHITECTURE" has been been assigned the following manuscript number: IJSEKE-D-19-00023.

You will be able to check on the progress of your paper by logging on to Editorial Manager as an author.
The URL is https://www.editorialmanager.com/ijseke/.

Thank you for submitting your work to this journal.

With kind regards

Int'l Journal of Software Engineering and Knowledge Engineering

In compliance with data protection regulations, please contact the publication office if you would like to have your personal information removed from the database.
--
Enviado via UCSMail.