

# dDocs: Data Analytics for a Citizen-Centric Platform

Daiane Oliveira  
Instituto Superior Técnico  
Universidade de Lisboa  
Lisbon, Portugal

daiane.oliveira@tecnico.ulisboa.pt

**Abstract**—With the exponential access over the information and communication technology, and the interesting in investing in applications still being higher in different areas, the Government is one of these interested areas, which results in the creation of a well-known concept: e-Government. This evolution decreases the bureaucracy, corruption and costs, while, increasing the citizens satisfaction since they do not need to go the physical locations and the end-process can be faster than the traditional way.

dDocs is a platform that was created having as a base the e-Government and, also, Record Management since documents needed to be changed from paper to digital. Having three user roles: citizens, curators and dDocs administrator, this platform provides the communication between citizens and curators without having the critical data.

These communications and, even, the user's management, results in events that could be transformed into something powerful, that is why it was decided to create a system to provide for each user role a dashboard to monitoring their activity, that is called dDocs/Dashboards.

In this dissertation, the dashboards were designed and developed after understanding the concepts behind it, analyzing the tools in the market that could be used during this work, understanding the best practices to reach a well-designed dashboard and collect the requirements. Also, it describes the process of evaluation to measure dDocs/Dashboard's quality.

## I. INTRODUCTION

Being e-government an intensive use of information technologies with the purpose of services provision, effectiveness improvements and democratic values promotion [1], which brings advantages, such as, time savings, costs savings, improvement of the public administration competence, decrease in corruption and implies less bureaucracy and Records Management, the field of management responsible for the efficient and systematic control of the creation, receipt, maintenance, use and disposition of records, including processes for capturing and maintaining evidence and information about business activities and transactions in the form of records [2], these two areas are capable of showing the importance of data, since they must deal with a huge amount of it, it must be careful with the data showed to the users.

Data visualization is an easy way for the human brain to understand what is behind in large or small datasets. The information is presented in a graphical form and it is quantitative. It can be used to make better decisions or to tell stories about the surrounding environment.

Between several data visualizations techniques, the most relevant one is a dashboard. It only has the most important

elements to tell a story and these elements are shown based on metrics, previously, defined to reach some goal to the user. It provides a way to combine data from different sources.

dDocs, a platform that supports the citizen-centric approach, where the citizen is the center of all interactions. The critical data does not pass through dDocs, it is directly sent to citizens, dDocs only has the document templates and the critical data metadata [13]. This platform is accessible to any device and allows to design electronic documents through orchestration of services and data provided by the curators and they can be managed. Also, it is a collaborative and social platform that facilitates and promotes the relationship between citizens and curators through any process that involves the request, publication, access, delivery and sharing of electronic documents. Any document can be searched by life events, by document types, by curators or by curator's classification [14].

In this platform there are three different permissions where the users can perform specific actions. The users permission, in cause, are citizens, curators and the administrator.

The citizen is someone that is not on dDocs as curator or administrator, he can search for a specific template and after select the desired one, it turns into a document. He is able to perform four actions: create, edit, close and, share a document with some curator or others citizens.

A curator is a public or private organization that, responsibly, designs and manages the templates owned by it.

Meanwhile, the administrator is someone that manages the dDocs system.

A document, in dDocs, has an identifier and it is created based on a document template and can have life events, which can be, for instance, birth, death, education, health or professional life and/or as document types, for instance, identification documents, certificates and requests for certificates.

In this platform, document templates are templates from the original document of the organizations without critical information and are managed by a specific curator, then, citizens can search them in dDocs.

The main features of citizens and curators, in the dDocs system [14], are:

### Citizens

- Access and management of his own documents;
- Documents sorting and search features;

- Document sharing with other users;
- Citizen document filling, feeding Curator data collection, or forms;
- Integration with payment systems;
- Hands free authentication mechanism;
- Several levels of legally binding authentication;
- Authenticity validation system for critical identification documents.

### Curators

- Curator-specific user management and role assignment;
- Management of protocols and contracts with other curators;
- Management and configuration of internal and external data services;
- Management and configuration of user interface objects;
- Management of document templates;
- Configuration and WYSIWYG interactive design of document templates.

dDocs was developed at the Information and Decision Support System Group of INESC-ID <sup>1</sup> and by the support of MDSS <sup>2</sup> that provide their fundamental concepts.

Being dDocs a system with events monitoring, where an event provides historical information related to it, making it possible to do something useful with this events and show the result to the users, which can improve users satisfaction in the system and, even, give an opportunity to them to track their activity.

Having the notion that this platform, dDocs, has three distinct user roles and events logs, there was an opportunity to design and develop a system of dashboards.

This paper has the following structure. Section II describes the background, analyzing over some technologies, such as, Data Visualization, Dashboards and Extract, Transform and Load tools. Section III introduces the dDocs/Dashboards requirements which are the technical constraints and proposed metrics to design the dashboards. Section IV describes the prototype. Section V discusses the dDocs/Dashboards based on two evaluations: performance and usability. At last, Section VI concludes the paper and exposes some future directions for this prototype.

## II. BACKGROUND

This section overviews the fundamental concepts to comprehend the context which this work is inserted in.

### A. Data Visualization

Before creating visualizations, there are some considerations to take into account. It is essential to understand who will be the audience, since this visualizations are a great way to communicate information, but if it does not meet the audience's expectation and needs, it can be a great disaster. Another consideration is about tools, first of all, it

is important to know what it is in the market and understand the differences between them to make the right choice.

Data visualization has been around for centuries in several areas of activity. In the late of 1700's, William Playfair, who is considered the "father" of data visualization, invented a few diagrams that are used often until today, like, bar charts, line charts and pie charts. Unfortunately, during his life, his work didn't have a major effect. The importance of representing information through visual was shown by John Snow when he decided to draw a map to determine the cause of the cholera epidemic of 1854. Later, Florence Nightingale, demonstrated, through pie charts, that the major cause of British soldiers deaths during the Crimean War was the lack of hygiene in the hospitals which improved the conditions of the hospitals. Then, in 1861, Charles Joseph Minard, mapped the march of Napoleon to Moscow to represent the massive loss of his army over the advance to Moscow and the retreat. Only in 1977, the Princeton University's professor, John Tukey, showed the data visualization techniques to analyze and present the data. Today, data visualization has an impact in BI and Business Analytics [11].

Visualization is defined as a communication of information through graphs. A single picture interpretation is processed at the same time as the human perception system, making information in pictures faster to understand than a page of words [8], for example. The speed of interpretation of a page of words depends of the sequence of reading. A graph or a map have the advantage of being understood by different cultures.

It is used on applications to provide different views of data and to detect some pattern or anomaly. Visualizations are an effective way for present information to people and enable to support some facts that happen, in real time, from global warming to economic trends [8]. Maps and charts are abstract representations of data that people see frequently and go beyond than being just for business and economic analysis. It could be used, for example, in MRIs and DNA sequencing results. In an interactive visualization it is possible to rearrange the information for preferences of users and an application can be fully driven through visualizations.

For constructing visualizations there are some guidance to think about before they are implemented:

- 1) Problem domain;
- 2) Data abstraction (What dataset will be used?, Where is it available?);
- 3) Task abstraction (What tasks?, What is the purpose?).

The tasks, described in [9], are:

**Discover** - Find new information;

**Present** - Transmit information, by telling a story with data, or guiding an audience through a series of cognitive operations;

**Enjoy** - Wake user curiosity that might be both stimulated and satisfied;

<sup>1</sup><https://www.inesc-id.pt>

<sup>2</sup><https://mdss.pt>

**Annotate** - New annotations in some visualization element that already exist;

**Record** - Save or capture visualization elements;

**Derive** - New elements based on existing ones;

**Lookup** - User search for information where s/he: knows what s/he is looking for and where it is;

**Browse** - User search for information where s/he: does not know what s/he is looking for, but know the location;

**Locate** - User search for information where s/he: knows what s/he is looking for and does not know the location;

**Explore** - User search for information where s/he does not know what and where it is.

**Identify** - Find information without prior knowledge;

**Compare** - Compare similar entities of data represented;

**Summarize** - Overview of everything.

## B. Dashboards

In BI, dashboard is the most popular data visualization technique [11]. It is defined by Stephen Few in [7] as a visual display of the most important information needed to achieve one or more objectives, consolidated and arranged on a single screen so the information can be monitored at a glance.

Dashboards have three functions to knowledge business performance: measure, monitor and manage.

As described in [10], it helps to decrease time to make decisions since it provides facts, through data. Also, concede the privilege to compare values since the past to grant new concrete objectives. It should focus on which data is more important to each role in the organization.

Dashboards can be classified in three categories:

**Strategic** - summarizes what is going on in the past performance of the organization.

**Operational** - tracks the organization's activities.

**Analytical** - allows interactions to explore the details of the data.

There are some challenges before the development of dashboards, one of them is to choose which metrics should be tracked. The dashboard design depends on the selected metrics. Furthermore, metrics helps to understand what will be achieved.

Basically, a metric can be defined as a quantitative data that may be counted, ordered or aggregated.

For instance, considering a CEO that wants to build a dashboard for his organization, what is the major metric? It could be net income. After that, consider what factors influence this metric, it could be sales, and, then, what factors take to sales? One reason could be because of the average sales price. As noticed, it is a good practice to think, first, at top-level and, then, lower the level.

About the design, some practices should be consider as describes in the next.

Firstly, the placement, attention, cognitive load and interactivity are important to improve the effectiveness of a dashboard.

Before beginning the construction of the design, fundamental questions should be answered to understand the need of develop a dashboard and, also, to know which information is necessary to show. For that, typically, it is essential to make interviews to question the following:

- 1) Who are the users?
- 2) What are their goals?

Several operations can be use in the dashboards, and they are described as:

- **Drill Down** - allows to give more details.
- **Filter** - shows only what is selected.
- **Comparison** - compares several data.
- **Alert** - prompts attentiveness when a metric is above or below of the threshold.
- **Export** - exports the data from the dashboard.
- **Summary** - details the description of a visualization.
- **Tagging** - allows to custom the dashboard with the data the user wants.
- **Annotation** - brief comments about a visualization.
- **Chart** - organizes and represents quantitative data. For instance, line charts, bar charts and pie charts.

Important information should be quickly noticed and the data visualizations should, clearly, show the information and must be easy to interpret.

Line charts and bar charts are a good example to visualize quick comparisons.

One important tip is about avoiding distractions, the dashboards should not contain animations, motion, too bright colors or too many colors.

On analytical dashboards, the use of interactivity should be essential. Filters and drill down are good practice to be in.

Meanwhile, on operational dashboards, contains several Key Performance Indicator, which is a way to measure if an action is in the right direction for attending some goal, purpose by the organization and they are used to compare and to alert, typically through colors (red when some KPI is below the threshold and green to inform that KPIs are in a good performance), the tonality of the colors should be always the same. It is important to notice that KPI is different from a metric, since the first one is an important indicator to reach some goal of the organization, while the second one is something to be measured. A metric can turn to be a KPI if is relevant for the organization's strategy.

Texts should be carefully chosen to be in the dashboards, only the important ones can be inserted and the size of them must not be small so, it won't difficult the reading. Usually, to understand if the use of texts are effective in dashboards, it is resorted to usability tests.

### III. REQUIREMENTS

This section describes technical constraints, which represent any of a number of technical issues and obstacles that impact the design and metrics, that are a set of quantitative measures to make comparisons, measurement or track performance.

#### A. Technical Constraints

To ensure that the final result of this work is compatible to the expectations of the involved people, it is important to determine the constraints. They are:

**Web-based solution:** The client application should be accessed on a web browser.

**Responsive:** The client application should be compatible with resolutions of different devices.

**Angular 2:** The client application development should be on Angular 2.

**MongoDB:** The data system should be kept in MongoDB database during the development.

#### B. Proposed Metrics

Based on the dDocs concepts described in section I and remembering that dDocs platform has three user roles: citizens, administrator and curators, it is important to highlight the relevant metrics for each of them to start designing the dashboards. These relevant metrics are:

##### Citizens

- **Number of documents by action over time (last month/last year/since ever):** Total of active documents by action from the citizen in time (last month/last year/since ever).
- **Number of documents over time (month/year):** Total of documents active from the citizen during the time (months/years).
- **Number of documents by document group (life event/document type):** Total of documents active related to a document group from the citizen.
- **Number of documents by curator:** Total of documents active related to a curator from the citizen.
- **Number of documents shared by the citizen over time:** Total of documents active that are shared to others over time.
- **Number of documents shared with the citizen over time:** Total of active documents that were shared to the citizen over time.
- **Number of documents shared with the citizen by a document group (life event/document type):** Total of active documents that were shared, having a document group, to the citizen.

- **Number of documents shared with the citizen by curator:** Total of active documents that were shared, related to a curator, with the citizen.
- **Number of documents shared by the citizen, by a document group (life event/document type):** Total of active documents that were shared by a citizen, having a document group (life event/document type).
- **Number of documents shared by the citizen, by curator:** Total of active documents that were shared by the citizen, related to a curator's document template.
- **Cumulative number of documents over time (month/year):** Total of cumulative documents active from the citizen during the time (months/years).
- **Cumulative number of documents shared by the citizen over time:** Total of cumulative documents active shared by the citizen over time.
- **Cumulative number of documents shared with the citizen over time:** Total of cumulative documents active shared with the citizen over time.

##### Administrator

- **Number of citizens:** Total of citizens that are active in the system.
- **Number of curators:** Total of curators that are active in the system.
- **Number of inactive curators:** Total of curators that are inactive in the system.
- **Number of templates:** Total of documents templates active in the system.
- **Number of inactive templates:** Total of documents templates that are inactive in the system.
- **Number of documents:** Total of documents that are active in the system.
- **Number of citizens over time:** Total of active citizens that were created over time, in the system.
- **Number of documents over time:** Total of documents that are active over time, in the system.
- **Number of templates over time:** Total of documents templates active over time, in the system.
- **Number of curators over time:** Total of active curators that were created over time, in the system.

- **Number of documents by action:** Total of active documents by action in the system.
- **Number of templates per curator in the current year:** Total of documents templates per curator, in the current year, that are in the system.
- **Number of templates per curator:** Total of documents templates per curator in the system.
- **Number of templates, per curator, over time:** Total of documents templates, per curator, over time, in the system.
- **Number of templates, per document group, per curator:** Total of documents templates, per document group (life event/document type), per curator, in the system.
- **Cumulative number of citizens over time:** Total of cumulative active citizens that were created over time, in the system.
- **Cumulative number of documents over time:** Total of cumulative active documents over time, in the system.
- **Cumulative number of curators over time:** Total of cumulative active curators over time, in the system.

#### Curators

- **Number of templates:** Total of active document templates that the curator has in the system.
- **Number of documents:** Total of documents created based on the curator's document templates.
- **Number of documents shares with citizens:** Total of documents that are shared in the system with other citizens using the curator's templates.
- **Number of documents shares with curators:** Total of documents that are shared in the system with other curators using the curator's templates.
- **Number of citizens served over time:** Total of citizens that used the curator's document templates over time.
- **Number of templates over time:** Total of created templates that are active, by the curator, over time.
- **Number of accesses by template:** Total of accesses of a curator's template, that is, the number of created documents, by the citizens, based on curator's document templates.
- **Cumulative number of citizens served over time:** Total of cumulative citizens that used the curator's templates over time.
- **Cumulative number of templates over time:** Total of cumulative templates that were created by the curator over time.

## IV. TECHNICAL DESCRIPTION

The current section reveals in the architecture, describes the data model and, last but not least, presents the design and implementation of the dDocs/Dashboards.

### A. Architecture Overview

Since it must be a web based solution, the architecture will be a client-server style.

Developed on MEAN<sup>3</sup>, which is a JavaScript full-stack framework that includes the following technologies:

**MongoDB**<sup>4</sup> - the database, which is the one that is required.

**Express**<sup>5</sup> - is a framework for Node.js that simplifies the way to build the server application. Here, it is possible to easily create a robust API since it provides HTTP methods and middleware.

**Angular 2**<sup>6</sup> - is an open source framework to write client-side web applications, using HTML where it is possible to extend its syntax and JavaScript MVC to be able to automatically synchronize data from a view. This framework was one of the requirements for this work.

**Node.js**<sup>7</sup> - asynchronous event driven JavaScript runtime for developing the server-side application.

The data visualization tool used in this prototype is amCharts JavaScript library. Meanwhile, Pentaho Data Integration is the ETL tool used during this work.

### B. Data Model

In the beginning of the process to accomplish dDocs/Dashboards there is a JSON file, here, each event has the following attributes:

- **Timestamp** - type: datetime. Shows the date and time of the event.
- **Object** - type: string. It describes the type of the object on each event was created. The options are: Document, User and Document Template.
- **ObjectID** - type: number. It is the identifier of the object and it is unique.
- **LifeEvent** - type: string. Represents the name of the life event, can be empty or have one or more events separated by comma, related to the object.
- **DocumentType** - type: string. It is described as the LifeEvent attribute but instead of representing the name of the life event, it has the name of the document type.
- **Action** - type: string. It is the action performed in the object. The options are, Created, Edited, Closed and Shared, when the object is a Document. Otherwise, the only option is to be empty.
- **OwnerID** - type: number. It is the identifier of the user that shared a document with the user who receives it.

<sup>3</sup><http://mean.io/>

<sup>4</sup><https://www.mongodb.com/>

<sup>5</sup><https://expressjs.com/>

<sup>6</sup><https://v2.angular.io/docs/ts/latest/>

<sup>7</sup><https://nodejs.org/en/>

If there is no shared document by other citizen, this attribute is null.

- **OwnerName** - type: string. It is the name of the user that shared the document with the user who receives it. If there is no shared document, this attribute is empty.
- **User** - type: number. It is the user identifier on each the object suffered a change.
- **IsAdmin** - type: boolean. Informs if the user is the dDocs administrator or not.
- **IsCurator** - type: boolean. Informs if the user represents a curator or not.
- **OrganizationName** - type: string. Describes the name of the curator.
- **OrganizationId** - type: number. Identifies the number of the curator.
- **Name** - type: string. Informs about the name of the user who created the event.
- **Password** - type: string. It is an encrypted password for logging into the system and is empty when the object is not a User.
- **State** - type: string. Represents the current state of the object in the event. The options are: Active or Inactive.
- **TemplateName** - type: string. It is the name of the template. It is not empty except when the object is a User.
- **DocumentName** - type: string. It is the name of the document. Only appears when the object is a Document.

The data processing starts by using a script, the JSON file data was inserted on a MongoDB collection. After that, through PDI, data suffered the necessary transformations and was inserted in other MongoDB database. Specifically, in collections. After all that, these collections were used for constructing several dashboards for different user permission. The table I presents which collections compose the dashboard for a citizen, the administrator and a curator.

TABLE I  
MONGODB COLLECTIONS USED IN EACH DASHBOARD

	Collections
<b>Citizen</b>	DocumentsActions LifeEvent DocumentType CuratorsMyShares LifeEventShared DocumentTypeShared CuratorsShared SharesReceived LifeEventSharedWithMe DocumentTypeSharedWithMe CuratorsSharedWithMe
<b>Administrator</b>	Citizens TotalCurators Documents Templates Totals
<b>Curator</b>	NTemplates AccessedTemplates NCitizensShares SharedWithCurators CitizensServed

### C. Design and Implementation

dDocs events is in a JSON file, then, consequentially, it is maintained in MongoDB NoSQL database and, after that, it is processed and clean up to be kept in MongoDB collections. Each collection has key-values about the data that were needed to show in the dashboards.

The data processing and cleaning is made on Pentaho Data Integration, which is a drag and drop environment that is responsible for Extract, Transform and Load (ETL) processes. It has some features like the capability of migrating data from different data sources, it is able to integrate applications, perform data cleaning and load a considerable amount of data.

In the server-side application, with Mongoose (MongoDB object modeling for Node.js) it is possible to query what is needed and then creating the API with the result of each query with Express.

In the client-side application, there is component that acts as a service that receives the content of HTTP page with the related REST API to create a function. Afterwards, these functions will be subscribed on the component of the application that will use the data. The components resort Underscore.js, jQuery, amCharts and they are constructed under Bootstrap.

### Citizen Interface

This interface has three tabs:

- My Documents, "Meus Documentos" - contains information related to his own documents.
- My Shares, "Minhas Partilhas" - shows information about the documents that have been shared by the citizen.
- Documents Shared With Me, "Documentos Partilhados Comigo" - displays information about the documents that are shared with the citizen.

Inside the My Documents/"Meus Documentos" tab, the citizen can get information about his number of documents by action - Created, Edited, Closed, Shared (with others) - from last month, last year and since ever. Below, there is a line chart that describes the number of documents owned by him through time, which gives two option: by year or by month. Each chart gives two informations: the number of documents at each date and the cumulative documents that exist till that date. The Citizen can be able to disable the information. Following, there is a set of donut charts where it is possible to explore the information related to each document group (life event, document type) and curators. In the document group the citizen selects if he wants to take a look at information related to life events or document types which shows the life event/document type name and the number of documents that the citizen has about this event or type. The curators donut chart means that the documents owned by the citizen were from some curator template. So, it contains the name of each curator with the related number of documents.

After selecting the My Shares/"Minhas Partilhas" tab, a line chart related to his shares is shown. So, a visualization of the number of documents shared by him at some date, as well as the cumulative documents that were shared by him until that date is shown. The citizen may disable each information and zoom it. Then, there are three donut charts where he can select which document group he would like to observe and another about curators, which count the number of documents shared by him related to some curator. After that, there is a table to show the top three citizens that received more documents shared by him.

The last tab, Documents Shared With Me/"Documentos Partilhados Comigo", has the same structure as the My Shares/"Minhas Partilhas" tab. The difference is that, instead of showing information related to his documents shares, it is about the documents that he received by others citizens.

#### *Administrator Interface*

Another role in the system is about the administration where someone from dDocs is responsible to manage the system and which implies that there must be a dashboard for this kind of user.

Initially, there are several overviews at the top that indicate the total number of citizens in dDocs, the total number of curators, the total number of inactive curators, total number of documents, total number of templates and the total number of inactive templates and the corresponding percentage of each that were created during the last week.

Below, a set of informations about the citizens is represented, which include a line chart for indicating the number of citizens that were created on the system at some date and in parallel there is a line that sums the number of citizens since the beginning until at some date that the administrator would like to know. Here, he can select what he wants to see, or the number of citizens through time, or the cumulative number of citizens, or both or none.

In this dashboard, the top four citizens that have more documents in the system are shown, where if the administrator hovers on the green area of each citizen, the percentage of documents that each have in the system appears.

As the citizens are responsible for creating documents, in this area, there is also another line chart but now the information is about the number of documents in the system at some time as well as the cumulative number of documents through time. The administrator can perform the same actions as the other line charts mentioned above.

Then, there is a table which contains the total number of documents distinguished between actions (created, edited, closed and shared) in the platform.

The next section is about the information related to curators, which have two line charts, one that represents the number of curators created at some date in the system and the other that represents the number of templates that were created (since only the curators can manage the templates) in the system across time, where in each chart there is the cumulative number of the respectively represented object.

Here, the administrator can manage what information he wants to see. The line chart, that represents templates over time, changes when the administrator selects a curator in the drop down in the templates per curator. The default data in this line chart is about all the curators. There are, also, two donut charts, one for letting him know which are the top 3 curators in the system, that is, which curators have more templates in dDocs and the other donut chart is about the curators that have templates at this current year, where the administrator can disable or enable which curators he would like to be in the donut chart.

Also, in the templates per curator, the administrator can select the curator, active in the system, that he would like to obtain an information, then, the total number of templates created by the selected curator appears. If he would like to obtain more detailed information about this number of templates, he can click on the More Info/"Ver Mais" button that displays the number of templates with the corresponding life event or document type.

At last, there is an overview information about the last added curator and the last inactive curator.

#### *Curator Interface*

Here, someone responsible for managing his organization will have, on top of this dashboard, an overview over the number of created templates by the curator, the number of documents created based on his templates, the number of documents that has been shared with other citizens and the number of documents that has been shared with other curators.

Then, the curator will have access to a line chart that relates the number of citizens that has been served by curator's templates and the dates where it happens. The line chart provides two informations: the number of active citizens created at some date and the cumulative number of citizens, in the same situation, until some date. If the curator won't take a look at some of these two informative option, he can simply disable the desired one. The chart provides the zooming functionality.

Another line chart is presented below to show the number of templates created through time as well as the cumulative number of templates at a date. Here, he can disable and enable what he wants to see and can zoom it the way he wants it.

Next, there is information about the access on his templates. In one hand, there is a donut chart that lets the curator know which of his templates have more access, that is, which templates were transformed into documents by the citizens. In the other hand, there is another donut chart that represents the templates that have less access. In both, the number of accesses in each template is specified. Also, the accesses on the templates can be visualized in a line chart. On here, the curator may choose what available template he wants to see the number of accesses over time. The default option is all templates.

In the bottom is located a list that has his five recently created templates. In each content of the list is the name of

the template and the respective date of creation. In this list, the most important is to understand which of them is the last created template, for this reason, it is highlighted.

*Discussion*

All these dashboard are responsive, that is, they are prepared to be visualized in different screen sizes of user’s devices and the language used was Portuguese for the fact that dDocs is a system where the audience is Portuguese.

Meanwhile, line charts are good for trends. Here, each line chart shows the evolution of information over time.

Donut charts show parts of a whole. The use of donut charts is to present the distribution of life events/document types/curators based on the number of documents/templates/accesses. For instance, in this visual there is a fast knowledge which has more documents, consequentially, also, knowledge of which has less documents.

V. EVALUATION

This section delineates two evaluations based on performance evaluation and usability evaluation. The performance evaluation shows the time that takes to load all data related to different dashboards while, usability evaluation presents the dashboards evaluation made by a few participants.

Furthermore, in which of these two kinds of tests is present all the aspects of the process that are involved.

A. Performance Evaluation

The performance of the different dashboards in the system is measured. To accomplish this measurement, it was resorted a script in C# that populates each MongoDB collection with a controlled random data (to make sense when analyzed it on dashboards), for a better performance was implement for each MongoDB collection indexes. This test has four scenarios:

- 1) 1000 BSON documents.
- 2) 10 000 BSON documents.
- 3) 100 000 BSON documents.
- 4) 1 000 000 BSON documents.

The results are separated by dashboards, namely, Citizen which contains three pages: My Documents/”Meus Documentos”, MyShares/”Minhas Partilhas” and Documents Shared With Me/”Documentos Partilhados Comigo”, Administrator and Curator.

Firstly, were tested a set of collections responsible for showing the information on My Documents/”Meus Documentos” for each scenario. The results are represented in the table II.

Analyzing the table II, the time that takes to load all the dashboard increases with the number of BSON documents in each collection. Even with 1 000 000 BSON documents, the time to load is good which proves that MongoDB is capable to handling very fast with a considerable number of data.

During the visualization of the loading time, in the My Documents/”Meus Documentos”, the table takes more time to show numbers after 100 000 BSON documents. The reason why can be in some calculations made in JavaScript.

TABLE II  
CITIZEN: MY DOCUMENTS/MEUS DOCUMENTOS PERFORMANCE RESULTS

Scenario	Time to Load (s)
1	1,87
2	2,03
3	2,17
4	2,34

In the next dashboard My Shares/”Minhas Partilhas”, the related collections were filled based on each scenario. Table III shows the results:

TABLE III  
CITIZEN: MYSHARES/MINHAS PARTILHAS PERFORMANCE RESULTS

Scenario	Time to Load (s)
1	1,71
2	1,75
3	1,80
4	1,97

The performance to load all the content in the dashboard is good. With 1 000 000 BSON documents took more time than expected, which could be explained by the fact that JavaScript math operations happen simultaneously.

The third tab in the citizen environment, Documents Shared With Me/”Documentos Partilhados Comigo”, all the related collections were prepared for the four scenarios and the results are presented in the table IV. The load time is similar as the other pages in citizen’s dashboard and the results are equal as the others.

TABLE IV  
CITIZEN: DOCUMENTS SHARED WITH ME/DOCUMENTOS PARTILHADOS COMIGO PERFORMANCE RESULTS

Scenario	Time to Load (s)
1	1,70
2	1,76
3	1,85
4	2,71

The table V shows the results related to the performance tests for the administrator’s dashboard. Here, one more time, the scenarios were done in the collections that belong to this dashboard. Until the third scenario, everything loads normally, afterwards, the line charts suffer a little clutter, since it has two y axis, the right one values were on the other axis line. Also, it took more time to load the content, when occurring the fourth test scenario.

The last dashboard that was tested for performance was about the curators. Testing in one of them, the load time results for each scenario are in the table VI.

The load time increases as the number of records in each related collection increases. Overall, it has a good

TABLE V  
ADMINISTRATOR PERFORMANCE RESULTS

Scenario	Time to Load (s)
1	1,81
2	1,87
3	2,05
4	2,48

TABLE VI  
CURATOR PERFORMANCE RESULTS

Scenario	Time to Load (s)
1	1,68
2	1,76
3	1,81
4	1,98

performance. The same as the other dashboards happens, that is, after 100 000 BSON documents inserted in each collection, the line charts suffer a clutter in the values of the y axis, also, took more time to load the data, the reason can be the calculations made in the client application to show some of the data.

In general, the performance of dashboards, when increasing the number of records in the MongoDB is linear.

### B. Usability Evaluation

To gathered information related with the user experience to this work, it was made a session where it was explained all the process until the performance test. Here, all the involved users took a look in all dashboards, where they were able to formulate questions and explore all the dashboards.

This session counted with the participation of four people, the lower number of users is explained by the reason that the understanding of this work is not easy and implied to combine a large number of inexperienced participants to be reunited in one session before the test to elucidate them about the scope where it is in. Also, Jeff Rubin and Dana Chisnell described in [15] that all participants should have nearly the identical experience. So, the participants are people involved in dDocs project.

In the end, to easily analyze the opinions, it was sent to each of these users a questionnaire to, briefly, answer seven questions:

- Q1.** How do you evaluate the dashboards?
- Q2.** How easy is it to understand the information presented in the dashboards?
- Q3.** How consistent are the dashboards?
- Q4.** How clean are the dashboards?
- Q5.** Do you think these dashboards are useful to dDocs platform?
- Q6.** What dashboard page or pages you like the most?
- Q7.** Do you have any suggestions?

The last question is optional, while the others are required to answer.

Table VII reveals the average score for the answers regarding the first four questions above, broken down by question. In overall, these questions had a positive score.

About the question 1, the participants think the dashboards are good. In the second question, dDocs/Dashboards is classified by three participants as extremely easy to use while one participant classified as easy. Meanwhile, in the third question, 75% of the participants think that the dashboards are consistent and 25% think they are very consistent. In the question 4, the answers were divided into clean and very clean.

TABLE VII  
QUESTIONNAIRE AVERAGE SCORE (1-5 SCALE) BY THE FIRST FOUR QUESTIONS

	Q1	Q2	Q3	Q4
<b>Average</b>	4.5	4.75	4.25	4.5

In the question 5: Do you think these dashboards are useful to dDocs platform?, the participants had to select one of the two options: yes or no and the results are that 100% of the users consider that the dashboards are useful to dDocs.

In the question 6: What dashboard page or pages you like the most?, the participants were enabled to select all dashboards, none or a specific dashboard page. Based on the results, the favorite dashboard is related to the curator (100% of the participants agree), the next preference is about the administrator's dashboard and 50% of them liked the citizen's dashboards. Actually, one of the participants selected his preference in all the dashboards.

The last question, which is related to give suggestions was answered by 50% of the users. Here the suggestions are: have incremental log imports to reduce the analyzed period and to change and, in the administrator's dashboard, from absolute numbers to percentage when distinguishing the number of documents per action.

To sum up, although the number of participants is small, these answers elucidate that this work received a good feedback.

## VI. CONCLUSION

Over time, the increasing use of platforms and the fast development of them, raise new concerns. One of them is related with the data, since years ago, there was an exponential increase of it that brought a lack of space and to solve it a well-known term, Big Data, appeared to allow large and complex datasets.

MongoDB is one of many existing NoSQL database systems. They are capable of supporting these large and complex datasets mentioned above. So, for that characteristic and, also, defined a requirement, during the development, MongoDB is used.

Also, e-Government and Record Management are fundamental areas that were merged to serve as a base to dDocs

platform. Being a system that provides communication with different users roles, the resulting data could be transformed into something valuable for the system.

Throughout history, data visualization had a huge impact and, nowadays, are commonly used in Business Intelligence and Business Analytics areas. Having the necessity to expose understandable information through visualizations, many tools have been created to support the implementation of the desirable data into many types of charts and maps.

Between the data visualization techniques, the most popular one is dashboard. One well-known definition is that dashboards are "a visual display of the most important information needed to achieve one or more objectives, consolidated and arranged on a single screen so the information can be monitored at a glance". Before starting to build dashboards, it is important to understand who is the audience and what they need, as well as what it is essential to define the most significant metrics to each role. Furthermore, there are some considerations to take into account related to the design of the dashboards, choose the right visualizations, choose the proper use of colors and an effective dashboard must be a responsive and contain an easy-to-read information.

In this dissertation, based on the resulting data from dDocs platform, is presented a set of dashboards each one for a user's role, which are: citizens, dDocs administrator and curators, taking into account the previous research and requirements. The process of development has passed by inserting the initial data in a MongoDB database, then, with Pentaho Data Integration the data was transformed to be stored in another MongoDB database. Having the conditions to proceed, the development of the dashboards with different technologies, being relevant the use of some JavaScript libraries, such as, underscore.js and amCharts.

This work has been evaluated in order to check its usefulness and quality and this evaluation took two strands, performance evaluation and usability evaluation. To get the results of performance test was made a script that filled each MongoDB collection with different scenarios followed by measuring the time to load the data in each dashboard and observe the dashboard's modification, in case they happen. The results of the usability test were captured through a user session where the participants explored the different dashboards and answered a questionnaire to assess the opinions of the participants over the look and consistency of the dashboards. In both evaluations, the results collected were positive and showed evidence that demonstrate the usefulness and feasibility of the data analytics on dDocs and support the thesis behind this dissertation. Moreover, the goals were accomplished.

The future research about this work can follow a few directions. Some ideas were appearing but reasons as time constraints or complexity do not turn to be implemented, it is important to notice that the main goals were accomplished in this work. Therefore, the future research directions could be:

**Incremental log imports:** this was one of the suggestions made in the usability test. This means, instead of processing

data since the first event of the log, when new events are inserted, this data should be in another log to be analyzed, resulting in a performance improvement in the ETL tool.

**Export data:** create a button at the top of each dashboard page to export data as .csv, for instance, for advanced users to manipulate the data for other purposes.

**Add a network graph:** In the administrator's dashboard could be implemented a network that shows the relation between the curators or even in the curator's dashboard to show the relation of his templates to the rest of the curators that may receive them as a document.

**Add a map:** In the administrator's dashboard for specifying the number of curators in each district of the country.

**Report:** add a new button in the dashboards to generate a pdf to save and have the report of user's dashboard.

## REFERENCES

- [1] F. Yao, A comparative analysis of administration-oriented and customer-oriented electronic government, in Management and Service Science (MASS), 2011 International Conference on. IEEE, 2011, pp. 13.
- [2] International Organization for Standardization, Information and documentation - Records management - Part 1: Concepts and principles, ISO 15489-1:2016.
- [3] A. Ribeiro, A. Silva, and A. R. da Silva, Data modeling and data analytics: a survey from a big data perspective, Journal of Software Engineering and Applications, vol. 8, no. 12, p. 617, 2015.
- [4] H. Ye, X. Cheng, M. Yuan, L. Xu, J. Gao, and C. Cheng, A survey of security and privacy in big data, in Communications and Information Technologies (ISCIT), 2016 16th International Symposium on. IEEE, 2016, pp. 268272.
- [5] Q. Xu, L. Yang, and C. Yu, Research on influence of big data to chinese e-government, in Reliability Systems Engineering (ICRSE), 2015 First International Conference on. IEEE, 2015, pp. 18.
- [6] M. Pannu, B. Gill, W. Tebb, and K. Yang, The impact of big data on government processes, in Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2016 IEEE 7th Annual. IEEE, 2016, pp. 15.
- [7] S. Few, Information Dashboard Design: The Effective Visual Communication of Data, 1st ed. O'Reilly Media, 2006.
- [8] M. O. Ward, G. Grinstein, and D. Keim, Interactive data visualization: foundations, techniques, and applications. CRC Press, 2010.
- [9] Tamara Munzner, Visualization Design and Analysis: Abstractions, Principles, and Methods. AK Peters, 2014.
- [10] A. Vaisman and E. Zimanyi, Data Warehouse Systems: Design and Implementation, 2014th ed. Springer, 2014.
- [11] O. Belo, H. Correia, P. Rodrigues, and R. Barros, A personalization system for data visualization platforms, in Innovative Computing Technology (INTECH), 2016 Sixth International Conference on. IEEE, 2016, pp. 162167.
- [12] T. A. Majchrzak, T. Jansen, and H. Kuchen, Efficiency evaluation of open source etl tools, in Proceedings of the 2011 ACM Symposium on Applied Computing. ACM, 2011, pp. 287294.
- [13] R. Santos, dDocs - A Citizen-Centric Electronic Document and Record Management System, MSc Thesis, Instituto Superior Técnico, 2016.
- [14] dDocs, MDSS, Citizen centric document technology, 6 2017.
- [15] J. Rubin and D. Chisnell, Handbook of usability testing: how to plan, design, and conduct effective tests. John Wiley and Sons, 2008.