TécnicoVis – Visualization of Patterns in Educational Data

Tiago Martins
INESC-ID and Instituto Superior Técnico, Universidade de Lisboa
Lisboa, Portugal
tiago.a.martins@tecnico.ulisboa.pt

Daniel Gonçalves
INESC-ID and Instituto Superior Técnico, Universidade de Lisboa
Lisboa, Portugal
daniel.goncalves@inesc-id.pt

Sandra Gama
INESC-ID and Instituto Superior Técnico, Universidade de Lisboa
Lisboa, Portugal
sandra.gama@tecnico.ulisboa.pt

Abstract—With the increase in the number of students worldwide, it has become difficult for teachers to track their students or even the institution itself to identify anomalies in degrees and courses. A viable option that allows following the evolution over the years, could be the analysis of academic data. However, analyzing such data without pre-processing can be a task that requires a high cognitive effort and time investment. So, a possible solution that allows overcoming this problem is interactive visualizations. In this work, we developed a system with multiple visualization techniques that allows the user to explore and analyze a huge dataset, with the purpose to follow the evolution over the years of all degrees and courses of Instituto Superior Técnico. Usability and utility tests were also conducted to evaluate the developed system.

Keywords—Academic data, Interactive visualizations, Evolution, Degrees, Courses

I. Introduction

Introducing new technologies in higher and distance education, like learning management systems (LMS), course management systems (CMS) and massive online open courses (MOOC), the number of students has increased throughout the years. Not only by the introduction of these systems, but also the increase of the world population of about five billion from 1950 to 2015 [8].

Using these platforms in teaching allows to easily store records related to student grades, attendance in classes and approvals in a course. Analyzing this data may allow detecting students with problems or even in the content of the course itself [2].

The need to analyze these data is related to the fact that the interaction between the student and the teacher is lower and students are more dispersed in all the possible courses in higher education compared to high school education, which makes it difficult to perceive their problems and to track down their path [9]. Regarding these data, it is possible to deliberate about the contents taught and to develop strategies to minimize failure and dropout. These strategies may imply a change in the contents taught in each course or a degree restructuring.

Given the amount of data that these systems allow storing of multiple years, schools, degrees or courses, analyzing them may be a very difficult task and a high cognitive effort is required. A possible solution that allows to understand and identify relevant patterns in data is information visualization techniques [6]. Monitoring the evolution of a degree or course over the years or discovering patterns and trends in the data would become a time-consuming process without prior data processing [1].

Information visualization allows to explore only the information that is needed to be analyzed, so it is possible to use filtering and selection mechanisms to reduce significantly the amount of data or through additional calculations, such as derived measures, that allow getting different perspectives from the original information. Static visualizations only allow the user to analyze the available information without being able to modify it, which makes it impossible to explore the data [10].

Regarding this, a system for Instituto Superior Técnico was developed, called FenixEdu, which allows the management of several academic components, such as students' grades, payments, schedules and other educational and administrative resources. The interaction with these resources generates data related to the students, courses and degrees, that can be accessed through a developed API. These data can be obtained by invoking methods, which can be public or private. However, the private ones allow the access to students' data, thus requiring their authorization [3].

In this solution, we developed a system that allows studying ways of analyzing the academic path and the evolution of degrees and courses over the years, with the purpose of evidencing information regarding the crossed data. This system may help the teachers, to avoid the analysis of several attributes in high sets of unprocessed data of their courses in different semesters. With these visualizations, it is possible to explore the data, using several mechanisms, like filtering and selection of attributes, to select only the relevant information. By doing this, it is possible to detect relevant patterns in the data and to overcome school failure.

In section two we explain the different works developed in the area, each exploring different mechanisms and academic data. The problems and strengths related to these works will also be discussed. In section three is presented the earlier phase of the implementation of our solution, which consists in the conception of the possible questions to which the visualization will answer and the description of the entire extraction process of the data. In the next section, we describe all the

978-1-5386-2080-9/17/$31.00 ©2017 IEEE

implementation phase, including the design of non-functional prototypes, the structure of the application and back-end and front-end development. In the fifth section, the method used to evaluate the tool will be described, namely usability tests and the case study carried out with the *Núcleo de Estatística e Prospetiva* of Instituto Superior Técnico. In section six is presented the conclusion of the work.

II. Related Work

The increase in the number of students in traditional and online education has provided a large amount of educational data with the introduction of learning systems. Analyzing student behavior and the evolution of courses and degrees has become easier.

Considering the analysis of the academic data, such as the academic path and the evolution of the courses and degrees, there was a need to develop applications that allow the teachers and the coordinators to keep improving the students' performance and overcome the academic failure. However, due to the increasing number of students in online teaching and the amount of data generated by them, it is hard to keep track of the students' difficulties and to help them overcome these issues, based on unprocessed data.

Considering this scenario MoocViz was created, which provides visual techniques that may help the teachers to evaluate the performance of their students. The visual representations used are the choropleth and the stacked bar chart. Choropleth summarizes the certificates delivered in all countries where the students are and it is represented by a white (fewer certificates) to red (more certificates) scale. The other representation allows the user to see a more detailed information, regarding the time dedicated to the degree. In this case, it's represented the access to books, exams, problems, lectures and tutorials. This allows the teachers to get a detailed view of what kind of content is visited by each country. However, due to the lack of interaction in both idioms, it's hard to explore all the available information [5].

Another application was developed to follow students' path across all the possible evaluations. Data from an English test of college students was collected and as an alternative to visual representations of bar and pie charts, parallel coordinates were used, to represent multiple dimensions. Each of those represents a different evaluation, where it is possible to track the evolution of the initial grade, until the final one. To avoid the visual confusion of the huge amount of gray lines, a range of colors (yellow to blue) was applied to distinguish the good grades from the bad ones. However, analyzing only one visual representation may not be enough, to get a detailed representation of the data, a box plot was implemented to analyze its distribution [7].

Despite this way of reporting information related to students, where both visual representations are independent and it's only possible to see one at a time, there is an application that has these requirements. It uses data of three years of the Faculdade Cenecista of Osório, which has information about the resources that students use, for instance, forums, web pages and mini-tests. To analyze data related to multiple students, a table lens was developed, so it is possible to compare all the different content accessed by each student and to represent their tendency to finish the degree. This attribute allows representing the students that might fail (green), that tend to give up (red) and those that might pass (blue). However, the colors used to represent this attribute, aren't intuitive at all since green was used to represent student failure, which might mislead the users. However, having a big number of students represented in the table lens, implies an equal number of bars, which may difficult its analysis without a proper filter mechanism. This application has also two more visual representations that are connected that work as overviews of the entire course. In the pie chart, is represented the number of students that accessed each resource and in the stacked bar chart is possible to see what accesses were made in a selected period. When the user checks some information in one of the views, the other is highlighted. When the period is changed, all the views are updated.

To allow the user to interact with the system, several types of mechanisms were developed, like the selection of each bar in the table lens to get more information about each student or even the ability to draw a rectangle to filter the dates present on the stacked bar chart. It is also possible to search for a specific student or a resource [13].

Based on the analysis of the different works, it is possible to extract relevant information about the visualizations used, such as the usage of filtering mechanisms and the selection of elements, to reduce the amount of visible data. By identifying the most common issues and strengths in the work referenced in this section, it is possible to cover those flaws and build an interactive and effective visualization.

III. PRE-PROCESSING EDUCATIONAL DATA

In the previous sections, the strengths and limitations of the applications and case studies that use visual representations were identified to help interpreting big educational datasets. So, to develop a tool that allows this analysis, it will be necessary to have access to this information. For this, a set of data of Instituto Superior Técnico, referring to all courses and degrees since 2002, is available using a developed API called FenixEdu. However, it is necessary to establish the information that users would like to know about.

A. Initial set of questions

With the high amount of data generated by this learning system, it is necessary to think about the most interesting questions, that may be answered. Considering the attributes mentioned in the related work, such as approvals, grades, class attendance, content visited and some attributes that may be extracted from the FenixEdu API, an initial set of representative questions was created, that will allow covering all the aspects considered important, whose answers could be obtained while interacting with the system. To evaluate some possible questions, we elaborated a form that was answered by 57 people, where the following questions were the most voted in a Likert scale:

- Has there been any improvement in the approvals of a course after a change of teachers?

- What is the degree where a common course has had more approvals?
- What is the course with the worst approval rating of a degree?
- Is there any correlation between the percentage of approvals and the evaluation of the CUQ reports of the courses?
- Which degree has the highest average number of course approvals?

B. Data Extraction

After analyzing the information that people would like to know, it was necessary to understand which attributes could be retrieved from the learning management system. After implementing a Python script to retrieve all the data, it was also sought to complement them with the evaluation of the teachers and courses, using the reports of the CUQ (Course Unit Quality) system, which consists in a way of evaluating all the courses of the university. With these reports, it's possible to obtain data after 2010, related to the medians of the evaluations submitted by the students regarding each course and teacher.

However, to avoid using unprocessed data, extra calculations were necessary to be made. Since it was not possible to retrieve information related to approvals before 2010, it was necessary to cross the data of each course of all semesters, to identify the disapprovals. However, this wasn't an easy task, due to problems with the data (wrong names and acronyms) and degrees' restructuring. With all the processing finished, the available attributes to use are:

- Approvals, grades and evaluation of the CUQ of each course and degree
- Courses' credits
- Number of enrollments in each course and degree
- Teachers' CUQ evaluations
- Average of Teachers' CUQ evaluations in each degree

Regarding the initial set of information gathered, where the data were unprocessed, it was necessary to transform them into valuable information that will allow answering all the identified questions, since it is necessary a lot of cognitive effort for the analysis of raw data.

IV. Implemented Solution

The focus of this solution is to allow a detailed analysis of several attributes, considering a temporal component of the situation of a degree and of all courses. Through the combination of several visual representations, it is possible to interact with them and explore educational data.

By doing this, it is possible to identify possible problems in education, so teachers and directive board can analyze the state of a degree or course without the need to use unprocessed data. By analyzing the visualizations, it may be possible to think about different contents to teach the students to increase the approval rating or even restructure the whole degree. The implementation phase was divided into two components, the back-end and the front-end represented in Figure 1.

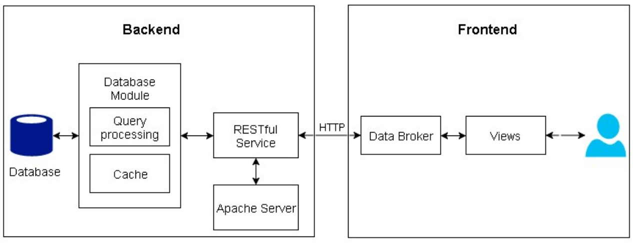

Fig. 1. Architecture

The technologies used to implement the back-end were a MySQL database, an Apache server, a data base module and a RESTful service written in Python. Regarding the front-end, the main technology used for the development of the customizable visual representations is the D3.js framework. This tool allows to manipulate elements in HTML and SVG and to assign styles in CSS, giving freedom to the development without using previously generated representations [11].

A. Back-end

Regarding the large amount of data collected, all the processed information was stored in a MySQL database, which contains the tables Degrees, Courses, and Teachers. In each table, there is the respective numerical identifier, the name, acronym, the CUQ of teachers and courses, grades, year, term, degree type and students number.

To make all these data accessible to the application, a module responsible for the processing of the queries was developed and a RESTful service that will allow the application to make the requests.

The module that connects to the database, has a cache mechanism that allows storing the data of the last twenty requests for each connected IP address. Since there is a cache associated with each address, only twenty requests were defined to be saved. Regarding the RESTful service, several endpoints were created to retrieve the data.

B. Low fidelity prototypes

Before the start of the implementation, some low fidelity prototypes were designed to illustrate the application's appearance and the visual representations.

Based on the defined questions, a representation was established to allow to compare courses and degrees simultaneously. The main visualization consists of an interactive matrix where each column represents the year and semester. Each cell would be filled with a range of colors from red (low approval rating) to green (high approval rating). However, as mentioned before, having only one visual representation isn't enough to analyze the different attributes and their relations. Considering this, a scatter plot was added that allows establishing correlations between the available attributes. To make the application interactive, a drop-down system was drawn to allow to filter each degree and course. An additional view was added to show the data related to the teachers that teach the courses. To represent the filtered course or degree, a pill was used with its name that can be removed when the user decides to.

C. *Matrix*

Considering the developed prototypes, we decided that the matrix would work as the main visual representation since it allows to represent all the available degrees and courses simultaneously. Each column of the matrix represents the years and semesters while the two first represent the names of the degrees and courses. Each cell contains the information related to the approvals of the students in the course. Rows without any values aren't drawn, so it is possible to avoid the increase of the size of the matrix. To represent the approvals, a scale of seven colors that follow the western color code (green for a high level of approvals and red to the opposite) was used. To establish the colors of the matrix, a quantiles scale present in D3.js was used. This allows handling the dataset in a discrete way, grouping the values of the domain in intervals, according to their distribution. When there isn't information in a semester, the cell won't have any color applied.

In a first iteration, the matrix was developed with all the courses represented on it, however; it made the idiom extensive. Regarding this, a new approach was developed, where only the degrees are represented in the initial state. To see the courses, it is needed to press the degree and it will expand to reveal them. Clicking again, will collapse and hide all the corresponding courses.

To avoid loading all the data of the degrees and courses in the initial state of the application is sent a request to get the information regarding the expanded degree.

The cells referring to the degrees were also modified, to contain the average of the approvals of all courses, to keep the view consistent. To allow a quick comparison between degrees or courses, an extra cell was added to each with the average of all semesters' approvals.

Although the degree expansion mechanism allows to drastically reduce the number of visible rows in the matrix, analyzing a specific row may be a hard task, so a filtering mechanism was developed, which consists of using a selection rectangle as identified in Figure 2.

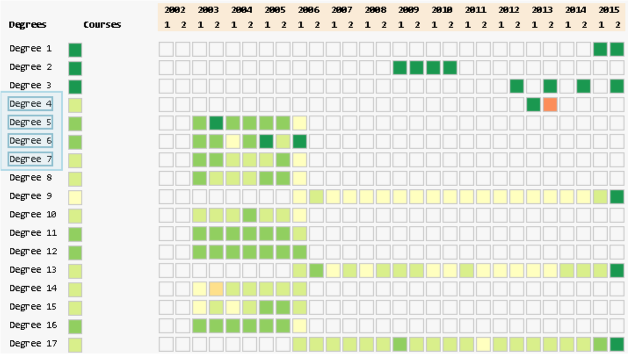

Fig. 2. Filtering mechanism

When the mouse is dragged next to the names of courses or degrees, the rectangle is drawn and updated according to the position of the cursor. To represent each filtered item, a pill was used.

To acquire more information regarding each cell of the matrix, a mouse over mechanism was implemented where it is possible to view the exact value of the approvals in that semester when the cursor overlaps the desired cell. Besides the extra information, a highlight mechanism was also implemented in each cell. The cell's limit is changed from gray to black. When a column or row is overlapped, a blue highlight is applied regarding the current row and column, so it is possible to read the year, term, degree and course selected (Figure 3).

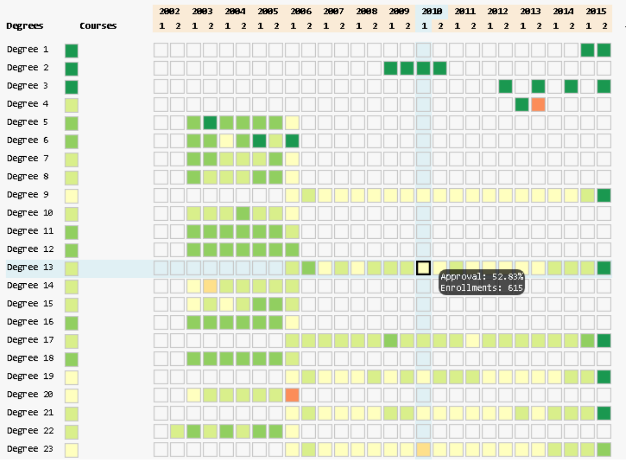

Fig. 3. Cell overlapping

Another mechanism was developed, and it consists of dragging a degree or course into a new position in the matrix, to allow the comparison between two rows. All that is needed is to drag the name with the cursor in a valid destination zone.

With all these mechanisms, if the user needs to filter a certain course, he needs to expand the degree and draw the selection rectangle. Based on this case, an alternative that would prove to be more intuitive and quick in the search for degrees and courses was developed. So, an additional filtering mechanism was developed, which consists of a search bar.

With some of the attributes referred, such as the approvals, the means of the students' grades and the CUQ of the courses, it was necessary to adapt the developed idiom to allow them to be chosen, so a menu was created.

D. *Scatter plot*

Although the matrix allows analyzing in detail the approvals, the evaluations of CUQ and the averages of the grades of all the courses and degrees over the years, it does not allow to visualize the possible correlations between the several attributes. So, an additional idiom has been developed to establish this comparison. Like the top of the matrix, the menu and the search mechanism, the scatter plot will also have a fixed position, so it is possible to go through the entire matrix and analyze the additional visual representation simultaneously. To avoid occlusion problems, an opacity of 0.3 was applied to each circle. The axes are adapted to the dataset which takes into account the maximum and minimum values of the domain.

Given a lot of data points, if the user wants to make comparisons between two distinct circles, it might be a difficult task. Given this problem, zoom and pan mechanisms were implemented that allow the user to explore this view. Regarding the zoom mechanism, it is possible to zoom in or out using the mouse scroll or double-click with the left button to make the sets of circles more scattered.

As for the pan mechanism, it is necessary to hold down the left button and move the mouse cursor to drag the dataset.

To select the different attributes, a menu was implemented in each of the axis labels. With the left mouse button, it is possible to press the labels, which triggers a similar action to the menu implemented in the matrix. Regarding degrees, the user can choose to view the average grades of all students, the average of the CUQ evaluations of all their courses and the total enrollment of students in all courses. As for the courses, an additional attribute was added, which consists in visualizing the number of credits.

Like the matrix, this representation also has a mouse over mechanism, to obtain more information, so whenever the cursor overlaps a circle, the tooltip is drawn and the item is highlighted in orange. In addition to highlighting, it was necessary to redesign the circle in order to overlap the others.

E. Line chart and Sparkline

To represent the teachers' CUQ evaluation, it was necessary to implement a third representation. Initially, we defined that multiple sparklines would be implemented in the lower zone of the scatter plot, with information for all degrees. However, this would be quite extensive, so we implemented together with the matrix. This graph serves as a summary of the expanded degree, where it is possible to visualize the mean of the CUQ of all the teachers in all courses in this degree over the years, thus allowing comparison with the present approvals in the matrix. Besides being possible to analyze the curvature of the line, points were added to help its interpretation, allowing to get more information on mouse over. An attribute selection mechanism identical to the scatter plot was also added. The user can choose to analyze the teachers' CUQ and the total number of enrollments in this degree.

However, since the matrix increases the number of rows with its expansion, it becomes difficult to make a comparison between the several degrees. To avoid this situation, a drag icon has been added, which allows moving the sparkline to the lower area of the scatter plot, turning it into a line chart for comparisons. To implement it, it was necessary to assign a color to each of the lines, so they could be distinguished. Whenever a degree is expanded and the sparkline is drawn, a random color is assigned to that degree that is recorded in an object, to ensure that there are no duplicate colors. However, comparing color strings doesn't guarantee visually distinct colors. To guarantee this situation, a formula was used to calculate the distance between two colors. Initially, it was considered applying the Euclidean distance to each color present in the object with the RGB color generated, however, the space of color L * a * b * allows to represent the colors visible to the human eye. The first formula developed to calculate the distance between two colors is the CIELAB Delta, which consists of applying the Euclidean distance to two arrays, with the following equation:

$$\Delta E = \sqrt{(L_2^* - L_1^*)^2 + (a_2^* - a_1^*)^2 + (b_2^* - b_1^*)^2} \tag{1}$$

However, to improve the above formula, the factors KL, KC, KH (factors of perception of the background conditions), SL, SC and SH (linear functions of the chromatic differences) were attributed. The values used for KL, Kc, KH, SL are one, while SC and SH have the value 1 + 0.045 * ($a_2 + a_1$) and 1 + 0.015 * ($b_2 + b_1$), respectively. The formula used to implement the difference between colors is:

$$\Delta E_{ab94}^* = \sqrt{\frac{(L_2^* - L_1^*)^2}{K_L S_L} + \frac{(a_2^* - a_1^*)^2}{K_C S_C} + \frac{(b_2^* - b_1^*)^2}{K_H S_H}} \tag{2}$$

To use this equation, it was necessary to convert the RGB colors to the color space L * a * b * and it is defined that values for ΔE greater than five, allow distinguishing if two colors are different [12]. To identify the rows that have been dragged into the comparison idiom, pills with the name of the selected item and the corresponding color are added. It is only possible to draw a total of five lines, to avoid a high cognitive effort in the analysis of the graph. Like in the sparkline, it is also possible to select the attribute and use the mouse over mechanism in each circle.

F. Multiple bar chart

The chart chosen to represent the detailed information related to teachers was the small multiples bar chart, where the height of the bar allows to encode the value of the teacher's CUQ, represented in Figure 4.

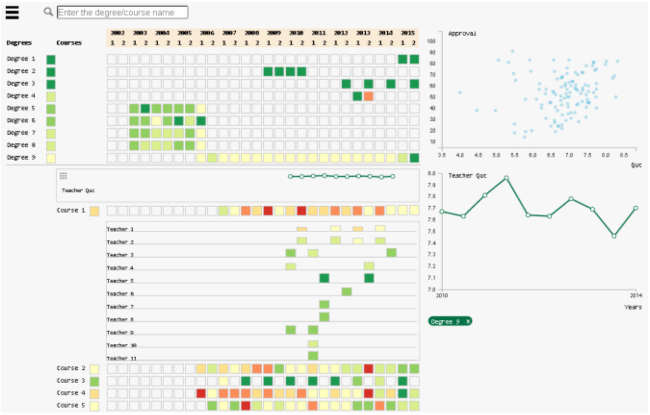

Fig. 4. Application screen with all the visual representations

However, since the values of the scores are similar, the color was added as second encoding. To keep consistency with the matrix, the same color scale was used for the values of the evaluations of each teacher. This view also allows using mouse over in each bar, to obtain more information. When there isn't any information about the teachers, this chart is not drawn.

G. *Interconnectivity between the visual representations*

To make possible to identify the items selected from the implemented views, a requirement was previously established that consisted of highlighting all selections. To do this, when the user overlaps the cursor in each element of the different views, it is reflected in the others. In addition, a selection mechanism has been developed that allows isolating the loaded instance with the left mouse button (Figure 5).

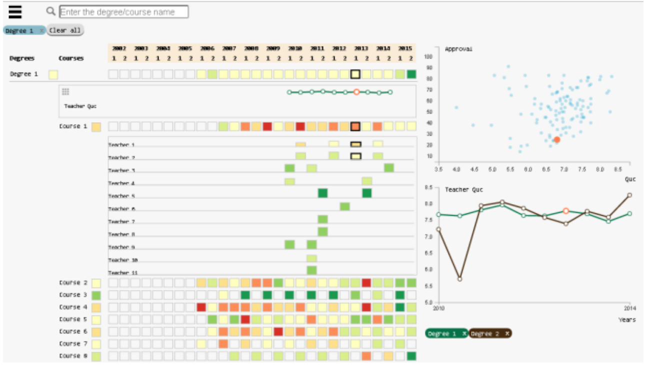

Fig. 5. Selection of an instance of a degree and a course

V. Evaluation

To evaluate the application, two types of tests were performed. The first is a usability test with several users, to understand if the system has intuitive mechanisms to analyze the data and if the codifications used to represent the attributes are easy to understand. The second test is a case study carried out with the *Núcleo de Estatística e Prospetiva*, where it was given the freedom to explore the entire application to reinforce potential usability problems and to see if the available information is easily understood, since this would be one of the target entities for the use of the developed system.

A. *Test protocol*

Experimental tests were conducted in a controlled environment directed to twenty people. These tests were performed in a room, to each of the users. The material used to perform the test was a computer to access the address where the application is hosted. Each test has a thirty minutes' duration and starts with a brief demonstration of the interaction mechanisms. Before starting the tasks, a trial period of the application of about five minutes was carried out, in order to allow an adaptation to the tool to execute the tasks.

The test has a total of eight tasks and while the users were trying to solve them, they described the actions they performed. To evaluate the interaction with the system, the following metrics were used:

- Number of errors
- Task completion time
- Completion of task

After completing all the tasks, each user answered to a satisfaction form (System Usability Scale) on a Likert scale, to evaluate the usability of the developed system [4]. To explore the different mechanisms, the following tasks were performed:

1) What was the degree with the highest approval average in all courses in 2008?

2) What is the degree where the AL course had more approvals?

3) What is the course with the worst grades average over the years of the LEMat degree?

4) Has there been any improvement in ACED course approvals in the LERC degree after a change of teachers?

5) Does the LEQ teacher's average CUQ tend to increase according to the percentage of approvals?

6) Of all graduations, which have more than 4000 enrollments and grades greater than 13.

7) Although LEIC-A has approximately twice as many LEIC-T students, do the courses' grades have big changes?

8) Has the number of teachers in the PEST course in LEMAT influenced the percentage of approvals?

These tasks allow identifying if the users understood all the developed mechanisms (sorting, filtering, summarization cells, attribute selection, degree expansion, pan, and zoom).

B. *Analysis of the results*

Based on the analysis of Table 1, it is possible to verify that both task 6 and 7 obtained a high standard deviation in the time used to perform them. They also have a longer average time than the rest. The main discrepancy is due to the lack of answer in the end of the four minutes of the task. However, it is also verified that task 5 has an equal number of errors because most users could finish it, but with wrong answers.

TABLE I.

	Time spent		Number of errors	
Tasks	Average	Standard deviation	Average	Standard deviation
Task 1	17.25	17.19	0.05	0.22
Task 2	29.8	13.07	0	0
Task 3	28.95	7.91	0.2	0.41
Task 4	36.8	11.21	0	0
Task 5	38.8	9.43	0.55	1.10
Task 6	105.5	64.65	1.05	1.85
Task 7	55,95	49.84	0.25	0.91
Task 8	32.1	9.20	0	0

During the execution of tasks 6 and 7, users who could not complete them, had some troubles while perceiving how to change the attributes in question. Although a visual hint was implemented when the mouse cursor is moved to a label, the users did not realize that the axis label is also a button and could be clicked. In question 6, many users made the mistake of searching in the search bar for the type of the degree and only then realized that this mechanism only makes it possible to search for acronyms.

Despite these errors, only one task has an average of over 1 error, which shows that most users understood the developed application. As for the time, it should be noted that only one task exceeded 60 seconds, which reveals that users needed less than a minute to answer a task that could take much longer by analyzing multiple graphs or tables without interactive mechanisms.

To prove this scenario for a broader population, statistical calculations were performed, in order to perceive the confidence interval relative to the mean. For this, a confidence interval of 95% was used. As such, we got the following intervals: IC_T1= [9.21; 25.29]; IC_T2 = [23.68, 35.92]; IC_T3 = [25.25; 32.65]; IC_T4 = [31.55; 42.05]; IC_T5 = [34.39; 43.21]; IC_T6 = [75.24; 135.76]; IC_T7 = [32.63; 79.27]; IC_T8 = [27.79, 36.41]. Given these intervals, it is possible to see that for 95% certainty values the first task does not exceed 25 seconds, the second, third and eighth do not exceed 37 seconds, the fourth and fifth do not exceed 45 seconds, the sixth does not exceed 136 and the seventh the 80 seconds.

As for the number of errors, the intervals are: IC_T1 = [-0.05; 0.15]; IC_T2 = 0; IC_T3 = [0.01; 0.39]; IC_T4 = 0; IC_T5 = [0.04; 1.06]; IC_T6 = [0.18, 1.92]; IC_T7 = [-0.18; 0.68]; IC_T3 = 0. Based on these intervals, it is possible to see that both task 1, 3 and 7 will have at most a value close to one error, while task 5 and 6 will have values close to two. As to the second, fourth and eighth it is not possible to calculate the interval because no errors have been made.

After performing all tasks, the users answered a satisfaction form and the final score was 75,63. According to the SUS score scale, this is a value that is situated on rank B, which reveals to be a good evaluation [4]. This way, it is possible to reckon that, based on the answers, the system has a good classification in terms of usability.

C. *Case study*

After performing usability tests and consequent annotation of all the problems identified by the users, some critical corrections were made. In this way, it is possible to carry out the case study, with the modified attribute selection and the attribute name of the CUQ. One change was to implement two additional visual hints that consist in drawing a bounding box around the axis label to alert the user that is in the presence of a button. Additionally, when the cursor overlaps the label, it is filled with a blue color.

As mentioned before, a case study was carried out with the responsible person of the *Núcleo de Estatística e Prospetiva* of Instituto Superior Técnico. This entity processes and maintains the data, with the purpose of conducting studies and present results related to school success.

Before starting the interaction with the system, all the developed mechanisms were explained. After exploring the entire application there were some suggested modifications to improve it. They consist in grouping pre-Bologna degrees, post-Bologna degrees, specialization degrees, master's degrees and integrated master's degrees.

As they are all together, the analysis by type of degree becomes a complicated task to accomplish. Another suggestion was to change the teachers' CUQ to a scale from red to blue, as it is used in the reports developed by this entity. Despite these suggestions, the responsible person was very interested during the interaction with the application, looking for scenarios that already knew of the reports produced and verified that, in fact, it was possible to see this information in the system. Using concrete scenarios from a course in a degree where the person already knew that there were weak approval ratings, the person used the filtering mechanism to limit the set of results and expanded the course to analyze the level of approvals over the years. Then the relation between approvals and grades on the scatter plot was established, selecting the attribute on the axis to confirm that it is indeed possible to easily analyze the information in a multi-year time component. An interactive tool that allows analyzing different attributes, such as approvals, CUQ evaluations, grades and number of enrollments, can give a huge contribution to the continuous reformulation of all curricular units, to improve the whole learning process.

D. *Discussion*

Regarding the purpose of the work developed and the requirements for implementing the application, it was intended that the tool could provide information related to the evolution of degrees and courses, considering a temporal component.

Based on the results of the usability tests and the case study where both proved to be very important to evaluate the developed system and meet what was defined as the basic aim of the development. Given the suggested modifications, some of them have already been implemented, such as the provision of a visual aid for the selection of attributes and the change of the attribute name for the CUQ evaluations of the courses. For future implementation, it may be necessary to group all the degrees according to their type, to modify the color scale of the CUQ of the teachers and to allow in the search bar the filter for the type of the degree. Following this, it is possible to improve all the interaction with the system, in order to make it more perceptible to the users.

Despite the many interaction mechanisms implemented, the participants, mostly, did not forget any of them and applied them whenever necessary, revealing that the initial demonstration of the system was sufficient for its use and that the application is intuitive and easy to learn.

VI. Conclusions and Future Work

With the increase of students in the academic area, it is important to follow their path and to analyze the adaptation of the courses and the degrees in relation to the current teaching. By collecting data, it is possible to monitor all these issues with the aid of interactive visualizations. Based on this context, data were extracted from Instituto Superior Técnico, which served as a basis to develop TécnicoVis, which allows analyzing the evolution of degrees and courses over the years.

In the first phase, several works were analyzed, where most don't allow curricular units monitorization. The problems and strengths in these works were also detected and it was based on this analysis that it was possible to choose the most relevant interaction mechanisms used in the application. Subsequently, it was necessary to proceed with the conception of the questions whose answers were intended to be obtained through the interaction with the application, followed by data extraction. After that, non-functional representations that met the initial aim were created. With the defined visualization scheme, the development phase of the system was started, which includes the implementation of the back-end and front-end.

In the case of visualizations, we used different types of views with alternative encodings. Regarding the matrix and the multiple bar graphs, the color range was used according to the western conventional code and for the sparklines, scatter plot and line charts the position was used as encoding, so it could be possible to give alternatives and intuitive ways to analyze the different types of data.

To evaluate the application, usability tests and a case study were performed, to ensure that it is easy to interact with the application through the implemented mechanisms that allow the exploration of the whole data and to highlight important patterns. Most of the tasks did not take over sixty seconds and with an average value of one error per task, which proves to be a satisfactory result and it is possible to conclude that the users perceived all the different mechanisms. Regarding the case study, it is important to note that given the context in which it is inserted, relevant patterns were identified during the interaction with TécnicoVis.

This way, it is possible to verify that the developed tool allows teachers to perceive and identify certain problems of the different degrees and courses and to act as an aid to improve teaching methods and consequently, to reduce course failure.

With usability tests, we realized that there are still some potential improvements to be made. Although some have already been implemented, it is still necessary to consider changing the colors of the CUQs to a scale from red to blue, to allow the search bar filtering mechanism to search by type of degree and group all its types. Since the current color scale isn't color blind safe, it will be necessary to include a scale that fits in this context in a future iteration of the system.

Besides all these changes, it would be interesting to have access to extra data that would allow focusing in detail on the path of the student. Have access to the courses' notes, tests and exam grades, high school grade and the gender of the students. In this way, it would be possible not only to analyze the evolution of degrees, courses and the more superficial analysis to the students (average grades) but also to analyze their situation in a more detailed and concise view. With access to this information, it would be possible to identify potential students with difficulties in certain courses and to develop educational content, in order to allow the student to achieve his potential throughout his academic career.

Acknowledgment

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC/50021/2013.

References

[1] Ali, L., Hatala, M., Gašević, D., & Jovanović, J. (2012). A qualitative evaluation of evolution of a learning analytics tool. *Computers & Education*, *58*(1), 470–489.

[2] Baepler, P., & Murdoch, C. J. (2010). Academic Analytics and Data Mining in Higher Education. *International Journal for the Scholarship of Teaching and Learning*, *4*(2), 1–9.

[3] Barata, R., Silva, S., Martinho, D., Cruz, L., & Guerra, L. (2013). Open APIs in Information Systems for Higher Education.

[4] Brooke, J. (2013). SUS : A Retrospective. *Journal of Usability Studies*, *8*(2), 29–40

[5] Dernoncourt, F., Taylor, C., O'Reilly, U.-M., Veeramachaneni, K., Wu, S., Do, C., & Halawa, S. (2013). MoocViz: A Large Scale, Open Access, Collaborative, Data Analytics Platform for MOOCs. *Nips '13*, (July 2015).

[6] Desai, A., Mian, M., Hazel, D., Teredesai, A., & Benner, G. (2014). Data visualization in educational datasets using a rule-based inference system. In *Proceedings - 2014 IEEE International Congress on Big Data, BigData Congress 2014* (pp. 462–469).

[7] Gan, X. (1), Liu, K. (2), & Liu, P. (3). (2009). Visual analysis of college students' scores in English test. In *Proceedings of 2009 4th International Conference on Computer Science and Education, ICCSE 2009* (pp. 1816–1819). (1)School of Foreign Language, Wuhan University of Science and Technology.

[8] Gao, Y. (2010). A study on mass higher education in the world-based on comparative perspectives. In *Education and Management Technology (ICEMT), 2010 International Conference on* (pp. 528–530).

[9] May, M., George, S., & Prévôt, P. (2011). TrAVis to enhance online tutoring and learning activities: Real-time visualization of students tracking data. *Interactive Technology and Smart Education*, *8*, 52–69.

[10] Mazza, R. (2009). Introduction to Information Visualization. *October*, *1*(978-1-84800-218-0), 139.

[11] *Murray, S. (2013). Interactive Data Visualization for the Web, An Introduction to Designing with D3*. Sebastopol: O'Reilly Media.

[12] Plataniotis, K. N., & Venetsanopoulos, A. N. (2000). *Color Image Processing and Applications*. New York, NY, USA: Springer-Verlag New York, Inc.

[13] Weiand, A., & Manssour, I. H. (2015). Towards visual analysis techniques for monitoring students of distance education courses. *6th Workshop on Visual Analytics, Information Visualization and Scientific Visualization, At Mercure Salvador Rio Vermelho Hotel. Salvador*, *4*, 5.