TÉCNICO LISBOA

Health Alert Dashboard

Health Alert Dashboard

Gustavo Afonso Teixeira Albuquerque Passos

Thesis to obtain the Master of Science Degree in

Engineering and Computer Science

Supervisor: Prof. Sandra Pereira Gama

Examination Committee

Chairperson: Prof. Paolo Romano
Supervisor: Prof. Sandra Pereira Gama
Member of the Committee: Prof. João Manuel Brisson Lopes

May 2018

Acknowledgments

I would first like to thank my thesis advisor Prof. Sandra Gama for having the door office always open whenever I ran into a trouble spot or had a question about my research or writing. She consistently allowed this dissertation to be my own work, but steered me in the right the direction whenever he thought I needed it.

I would also like to thank the experts who were involved in the validation survey for this research project: Prof. Hugo Nicolau without their passionate participation and input, the validation survey could not have been successfully conducted.

I would like to thank my friends for accepting nothing less than excellence from me, my family: my parents and to my brother and sister for supporting me spiritually throughout writing this thesis and my life in general.

Finally, I must express my very profound gratitude to my friends Carina Fonseca, Nuno Nogueira and Pedro Dias for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them.

To each and every one of you – Thank you.

Abstract

Medical records have long been vital tools in patient care, and current technologies are bringing medical records into the 21st century through innovative software and hardware computer programs. Nowadays, those medical record are dispersed between paper and digital format making its analysis very exhausting and problematic. Trough time many research of medical data visualization was made and we will discuss some of the existing software and applications so we can understand what was already done. The main goals of these research, based on information visualization, is to innovate and make the data more user friendly so health care experts make better use of the data. Also, based on the potential of information visualization, we created a web application, a interactive dashboard, to make easier the information access to health care experts like doctors, nurses allowing the analyzing it as a whole. Users tests show that with the dashboard, information retrieved is faster than the current softwares and data structure.

Keywords

Data Visualization; Innovate; Web Application; Dashboard;

Resumo

Os registos médicos são desde á muito tempo, uma ferramenta importante para o tratamento dos pacientes e neste momento, a tecnologia está a trazer os registos médicos para o século XXI através de software e hardware inovadores. Hoje em dia estes registos médicos estão dispersos em papel, electronicamente por softwares diferentes e por vezes desatualizados. Este facto faz com que a pesquisa e análise muito difícil e desgastante. Ao longo do tempo foram feitas diversas pesquisas sobre visualização de informação na saúde e iremos discutir alguns softwares e aplicações já existentes para perceber o que já foi feito nesta área. O objetivo principal desta pesquisa, com base na visualização de informação, é inovar e fazer com que os dados sejam mais acessíveis para todos os médicos para que seja usada mais eficazmente. Com o potencial da visualização de informação criámos uma aplicação web, um dashboard interativo, para facilitar o acesso á informação a todos os especialistas nas diferentes áreas da saúde na análise dos dados como um todo. Testes com utilizadores mostram que, com o dashboard, a informação é obtida mais rapidamente do que com as atuais ferramentas usadas pelos mesmos.

Palavras Chave

Visualização de informação; Inovação; Aplicação web; Dashboard;

Contents

List of Figures

List of Tables

1

Introduction

Contents

Medical data are often large, complex, disorganized, and may reside in separate databases or in data warehouses. Medical data are also messy, incomplete, and may contain systematic errors that prevent analysis. These characteristics make medical information difficult to integrate and analyze. However, critical analysis of medical data can improve health care and have a positive impact on lives. Information visualization is one way to improve understanding complex data and consequently increase the value of electronically available medical data so effective information visualization techniques can both reduce the problems medical analysts face and ease their analysis tasks [18].

"Computer-based visualization systems" can be defined as "visual representations of datasets intended to help people carry out some task more effectively" [19]. Information visualization techniques focus on datasets with non spatial data attributes and discrete observations. Scientific visualization, on the other hand, concentrates on visualizing real objects with spatial dimensions typically three-dimensional (3D), for example, tumor nodes or blood vessels. With proper visualization we are able to map abstract data to compact representation to convey meaningful information quickly. Furthermore, interactive information visualization is one way to enable exploratory analysis, an important aid to statistical confirmatory analysis.

Analyzing a single patient and analyzing multiple patients are different goals. The difference creates a dichotomy in the visualization systems. All systems in this survey either support tasks for analyzing a single patient, or those for analyzing multiple patients. Tasks that involve both, such as comparing a single patient with multiple patients with a similar history, are not extensively explored. Likewise, changing from multiple patient analysis to single patient analysis and vice-versa is not widely studied.

1.1 Objectives

The main goal of this research is to study ways to visualize medical data in order to make relevant information available to the user while allowing pattern discovery.

- Study approaches to information visualization in health care.
- Learn the technologies behind the development of a dashboard with multiple visualizations.
- Create a dashboard with multiple visualization in order to help health care professionals, dentists, to prescript medications better, to plan treatments more effectively and improve patients health conditions.

To do so, several user tests were made. Specific tasks and questions were created to help stay focused on users requirements. The data was provided by the research work from an other colleague [20]. With data cleaning tools data will be analyzed and processed to make better visualizations. The goal of all the tasks and processed data is to get the best prototypes (low fidelity and functional) to finally get the desired visualization.

.

1.2 Document Structure

This document is organized as follows:

Chapter 2 concerns *Related Work* where multiple solutions that already exist are explored and discussed.

In chapter 3 the *Dashboard Alert* is presented, in which the architecture and its components are presented, as well as the project's approach.

In chapter 4 we have *Evaluation* in which the evaluation for this project are presented and discussed including usability tests and its results.

Finally in chapter 5 the *Conclusion and future work* summarizes all the current conclusions of this work and the next steps to be done.

2

Related work

Contents

A lot of research has been made in information visualization of medical data because of its complexity and abundance.

In general we have three kinds of fundamental approaches: population level disease control, single patients and multiple patients using different paradigms. In this chapter we will discuss some of them.

2.1 Population level disease Control

Disease Data Correlator [1], is a visualization with different idioms that will help to have better comparisons.

Health is a common subject for all. So a tool that allows people to search and discover should provide different views depending on user's objectives. Some of these representations can some times prevent epidemic events.

The map view is a Choropleth map. The map in figure 2.1 shows the amount of people having a particular disease encoded a six color sequential color map in which darker represent a high percentages of people infected and lighter colors represents lower percentage. The detail view is dynamically loaded when a user selects a region on the Choropleth map.

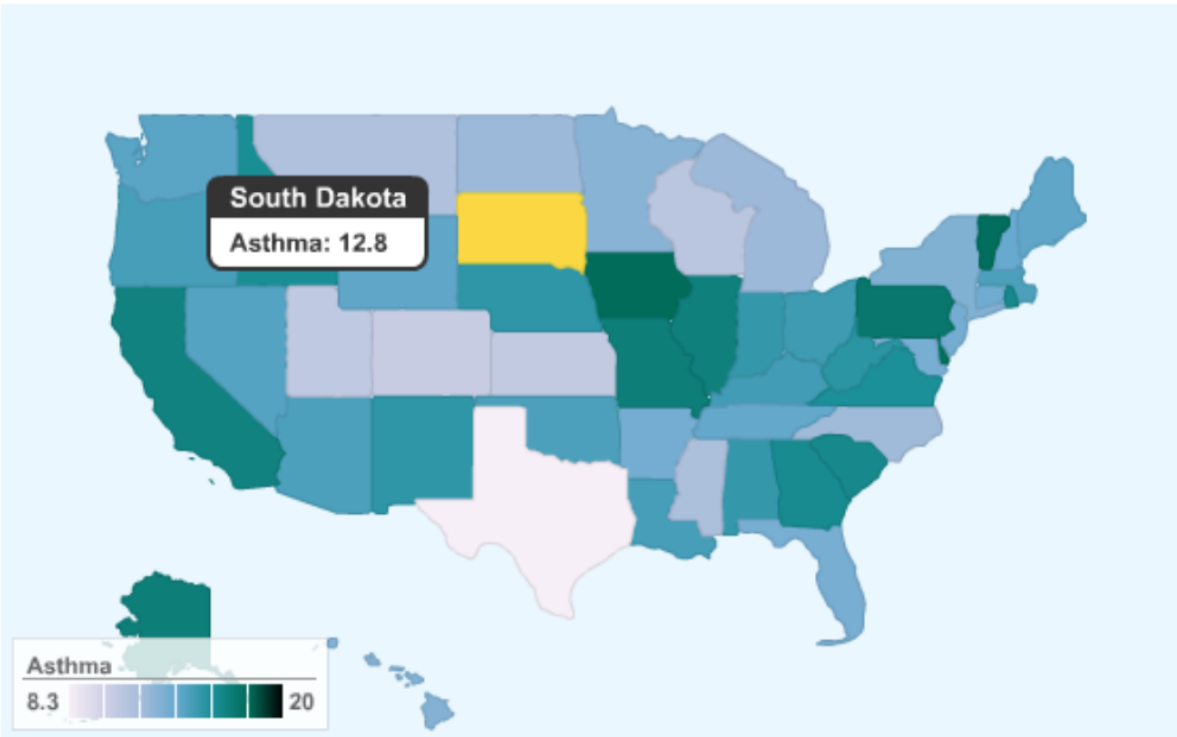

Figure 2.1: The Choropleth map in the Map View showing the state-wise distribution of the Asthma population of US [1].

With the data available several charts are created. For Race-wise prevalence and gender-wise prevalence,a 3D bar chart is used. According to Visualization Analysis and Design [18], a 3D bar chart is not the best way to represent data. It introduces unnecessary errors compared to 2D bar charts. The other are line charts that seems to have the correct use.

The figure 2.2 shows the correlated view of the demographic data to all the states in a single chart. A problem of this representation is that to show a clear picture of the state wise prevalence data the charts need to have a longer range along the X axis to plot all the states along the axis.

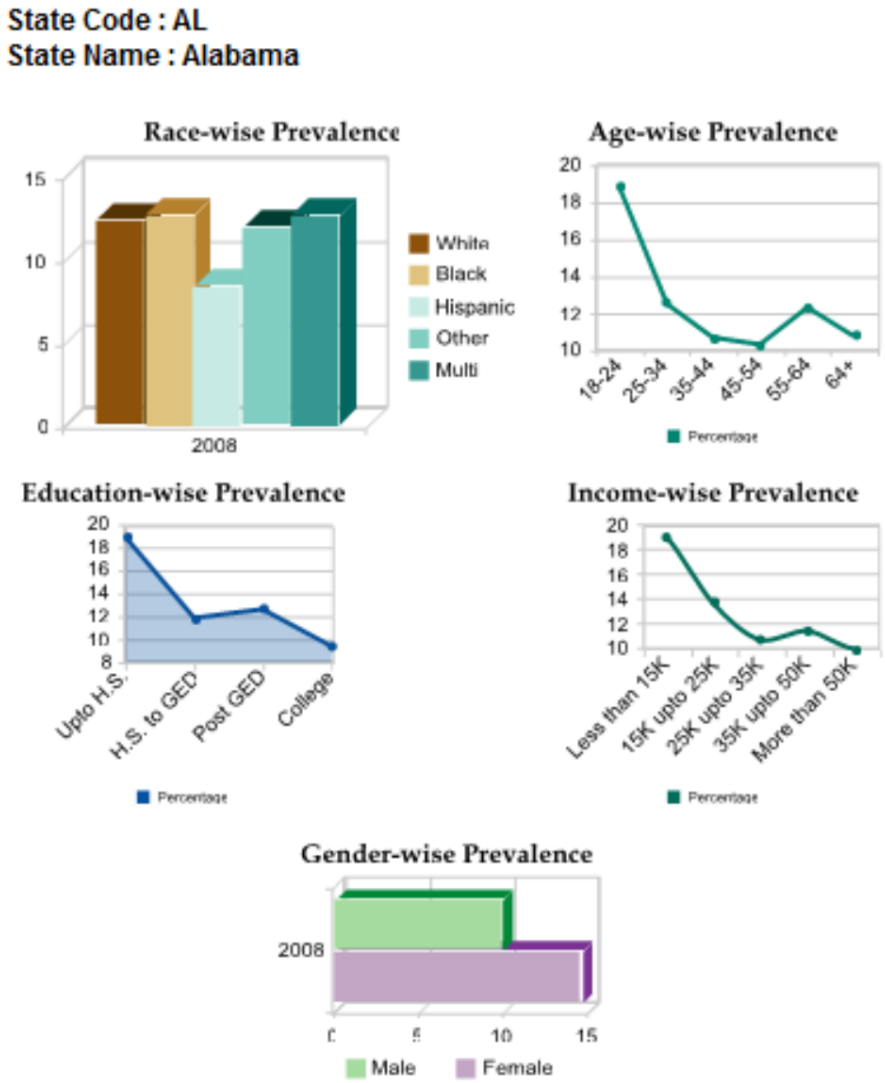

Figure 2.2: Details view window loaded with the demographic group-based data for the selected state [1].

To control trends and how to relate all the data to have accurate information Analytic Injury Dashboard [2] has been created by the Canadian Hospitals Injury Reporting and Prevention Program (CHIRPP)[1].

The types of visualizations (bar chart, line chart and buble chart), were selected to efficiently illustrate trends and patterns in injury data. As we can see on figure 2.3 the system consists of a single screen display of multidimensional information including temporarily and demographics related to one particular injury issue. Each window view is connected to a full-page visualization with specific functionalities and features to integrate analytical models and ease data exploration and analysis. Every time the user selects a particular graphical display, a comprehensive visualization will be displayed with advanced analysis capabilities and granularities to enable interaction and in-depth understanding of key injury indicators' performance. The data analysis component of the AID dashboard also enables users to interact with the visual representations at various levels of granularities using advanced visualization techniques (i.e. interactive distortion, zooming, filtering, brushing and linking). This interaction eases data exploration while minimizing cognitive load.

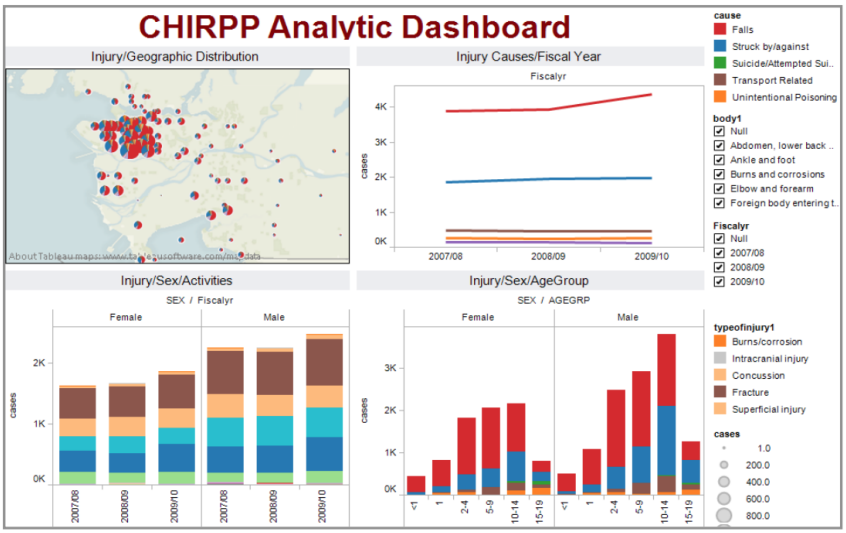

Figure 2.3: The AID [2] dashboard showing the CHIRPP injury data. It incorporates 4 multiple views: the geospatial visualization, the time trend visualization, and the 2 stacked bar charts visualizations.

[1] https://www.canada.ca/en/public-health/services/injury-prevention/canadian-hospitals-injury-reporting-prevention-program.html

With the same paradigm, RODS [3]- Real-time Outbreak and Disease Surveillance, is a simple and clean user interface, which is used for the detection of early disease outbreak. What this system exactly does is that it receives real-time data from hospitals over leased lines of health protocols. Univariate and multivariate statistical algorithms detect any abnormalities in the patterns of syndrome counts. In figure 2.4 we can see a line chart of sample data. The red line indicates the threshold level at which an alert would be generated; the blue line indicates respiratory syndromic counts from emergency departments of participating hospitals; the yellow line shows laboratory confirmed cases.

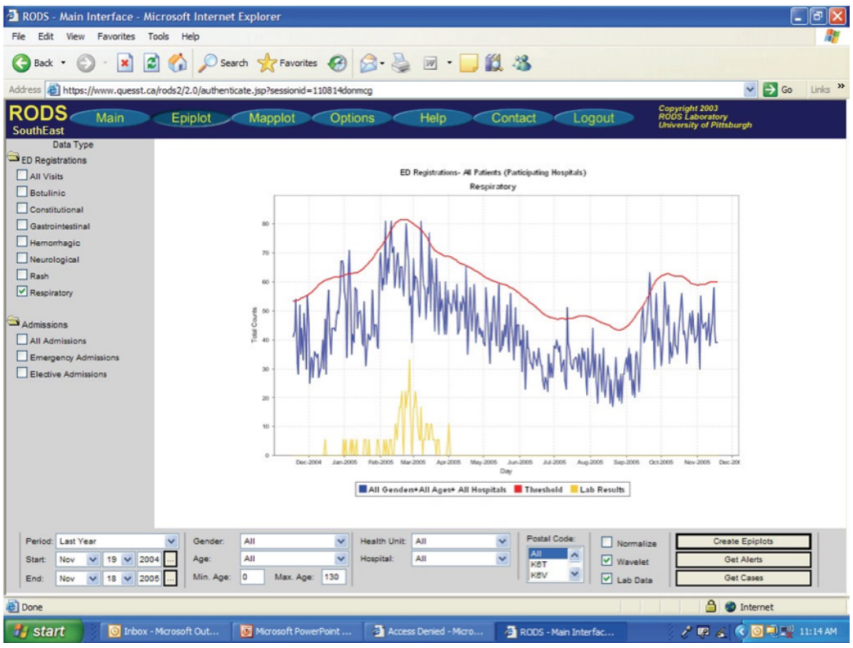

Figure 2.4: Example of function of sample Data – threshold, syndromic counts, and laboratory confirmed cases [3].

2.2 Single Patient

LifeLines [4] is an important paradigm, having inspired other research. LifeLines [4], as we can see on figure 2.5 supports the visualization of many medical variables on the same screen and displays the medical record, summarized as a set of lines and events on a zoom-able timeline. Aspects of the record are grouped by problems, allergies, diagnose, etc. When a problem or other condition becomes inactive the line stops. Color can be used to indicate severity or type.

The user can do the following when visualizing a patient record:

Details on demand - All events visible on the display form a menu giving rapid access to detailed information (triggered by a double click on the event.) The detailed information appears in a separate page covering part of the display or optimally in tiled windows on the side.

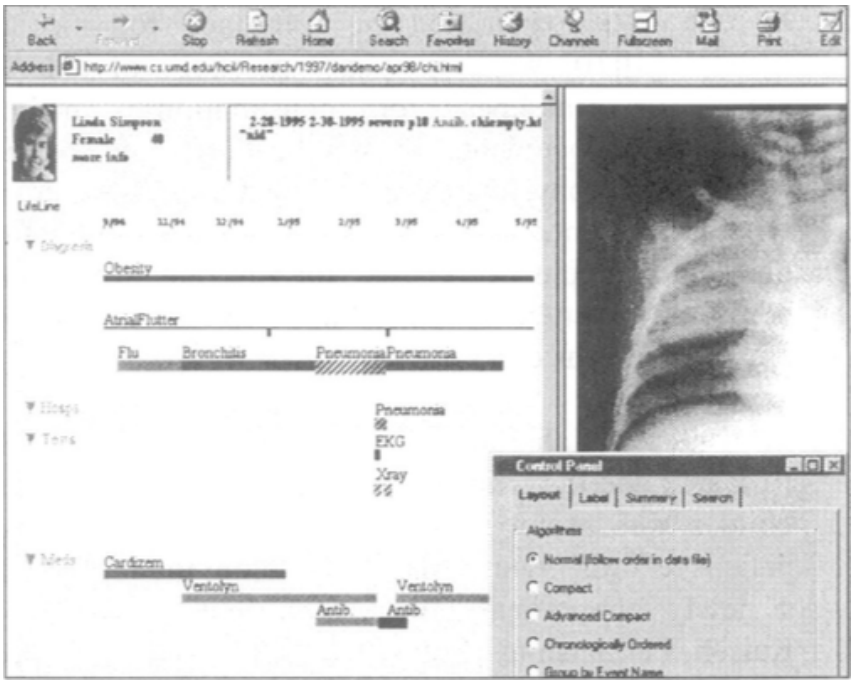

Figure 2.5: Example of patient health record in LifeLines [4]

Zooming - Zooming in and out can be done either by using the zoom-and-pan slider at the bottom of the display, or by clicking on the background of the image near the events that should end-up in the center of the zoomed image. A right button click on the background zooms out again. Zooming in and out reveals different levels of detail.

Highlighting relationships - In addition to the implicit vertical and horizontal relationships, searches can be performed on the entire record, highlighting all parts of the record that match with the selected region.

Coding attributes - Control panels are available to setup the mapping of the main display attributes (label, color and line thickness) to the data attributes.

Summarizing - Another way to deal with large records is to allow summarization of the events. The architecture allows a set of events to be recursively aggregated and replaced with summary events.

With a different level of interaction, we have VisuExplore [5].

It stands out due to its flexibility and task-specific interactions for the exploration process. The exploration designs only work with either quantitative or nominal data and rely on a single visualization technique that shows development over time. This includes the flexible arrangement of variables, smooth zooming and re-sizing, measuring time spans, and keeping track of time and relevant items. The design excels in allowing users to interact with variables. In figure 2.6 multiple groups of variables are visualized with different techniques in a multiple views display: Line plot (v1), bar chart (v2), event chart (v3), and

timeline chart (v4). All visualizations share a common time axis (v0). Patient master data is printed above the time axis. Users can interact with the visualization through mouse actions inside each view, scrollbars, a toolbar for time (i1), two toolbars for views (i2, i3), and the menu. When hovering over an item, a tooltip (i4) displays the item's exact time and value. A mouse tracker (i5) marks the position of the mouse pointer with an orange line to keep users informed about items occurring together and their time of occurrence.

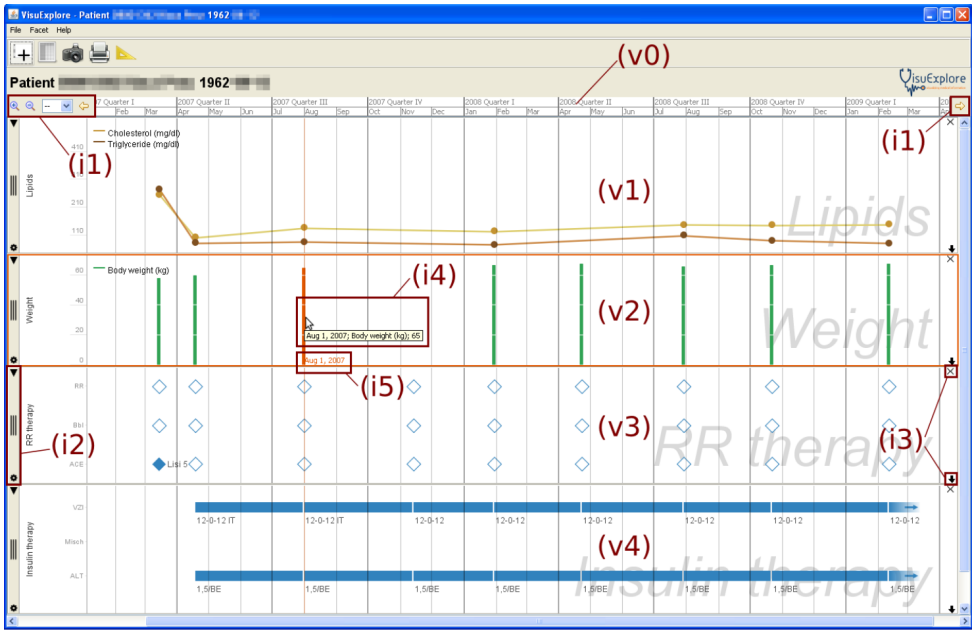

Figure 2.6: Exploring medical data of a patient using the VisuExplore [5] interface.

Users can open an additional window showing a table containing the items of one diagram. The table window allows users to inspect all items in detail. Users can select items by clicking on them. The diagram and table are linked: selecting an item in the table will select the corresponding item in the diagram. Here, in the diagram, the user has selected the four latest readings in "Lipids".

To add a new diagram users just need to select variables, a technique, and optionally a diagram title. Users can also move a diagram by clicking and dragging. While moving, the mouse drags a semi-transparent preview of the diagram and a blue bar marks the current insert location.

A different method to gather medical data is via tele-medicine, the use of telecommunication and information technology to provide clinical health care at distance. We have Caregiver [6] to visualize that kind of data.

One of the key concepts of information visualization is focus and context, the idea of embedding different scales in a single view in order to maintain context while navigating and examining details. Caregiver [6] presents focus and context techniques in two ways. Horizontally, selected areas can be enlarged in time by sliding a "magnifying lens" along the time axis. The region inside the lens is enlarged,

while the two regions on the outside are compressed accordingly. This provides a very efficient and intuitive mechanism to navigate and find details on a large time-scale, without losing the orientation or overview. Vertically, patient details can be enlarged by double-clicking on a patient timeline in the overview. The chosen time line expands dynamically to reveal full details about selected attributes. The other timelines are compressed accordingly to make space for the expanded panel, but remain visible at all times.

In figure 2.7, for example, on the left side there are two views that show the data as timelines. The goal of these views is to present a single overview of the whole data in a graphical way, while allowing quick access to details of interest. Timelines are an intuitive way to show the data for the current state of the patients related to past developments and the interventions that were administered.

The upper view shows the duration and size (number of patients mapped to thickness of the line) of the patient peer groups formed for the purpose of common interventions. Clicking on one of the groups highlights the participating patients in the lower view, and vice versa. The lower view shows one timeline for each patient in the program. One of the attributes can be chosen to be shown in this overview.

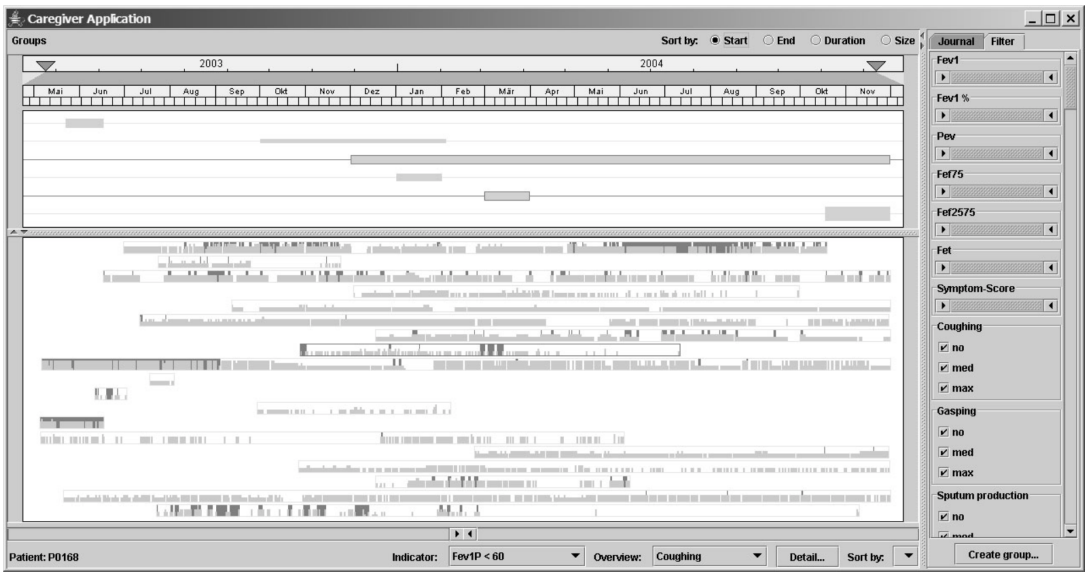

Figure 2.7: Example of Caregiver [6] interface

Following the same arrangement of idioms, with multiple line charts, we have MIVA, the Medical Information Visualization Assistant [7], formerly known as Critical Care Patient Data Visualization system. It uses point plots to visualize change of numerical values over time. Each variable is displayed in a separate plot. As we can see in figure 2.8 a gray band in the background denotes the normal range of the variable in each plot. The point plots are stacked vertically and aligned on a shared time axis that users can zoom and pan. The small plots on the right shows always the most recent developments and the current value is displayed as a large label. The variable label is color coded to indicate recent

abnormal values. Variables can be added from a list or reordered by drag and drop. Users can display all variable values at a certain time through the “scrubber”, which is represented as a vertical red line. On the top of the main page, there are two small panels with clinical text notes, clinical interventions, and other medical events.

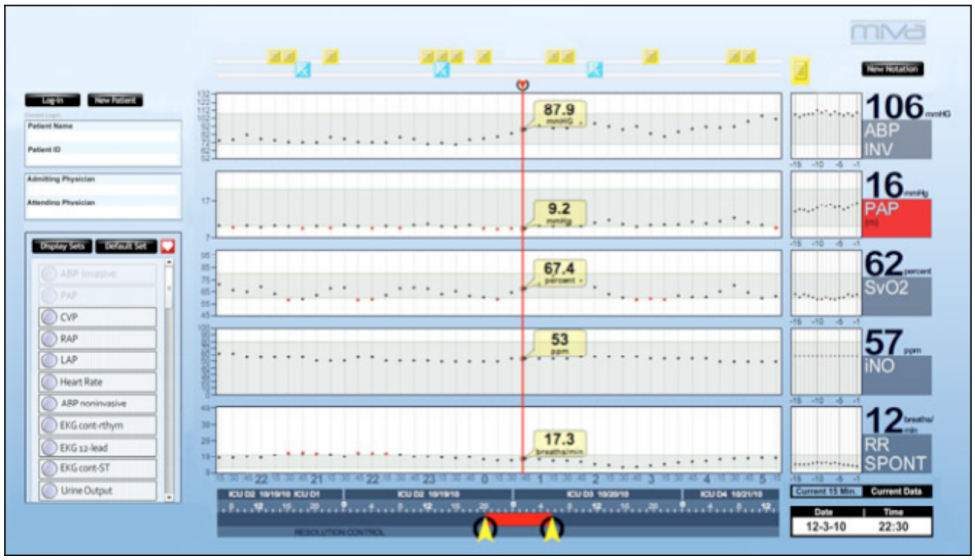

Figure 2.8: Example of MIVA [7] interface with control panel and data visualizations. Note the clinical and intervention notes entered by clinicians at top, designated with icons above the visualization display.

With different user interaction we have Midgaard [8]. The goal of the Midgaard project is to present different interactive visualization techniques that enable users to reveal the data at several levels of detail and abstraction, ranging from a broad overview to the fine structure and introduce a time visualization and navigation technique that connects overview+detail, pan+zoom, and focus+context features to one powerful time-browser. Midgaard can also be used to visualize patient data in combination with treatment plans (indicated by gray areas in the background of the patient data, as we can see in figure 2.9) and it provides different views for physicians, nursing staff, and patients. About interaction it is possible to do a semantic zoom, browsing through different visualization techniques and data abstraction levels from a broad overview to a fine structure.

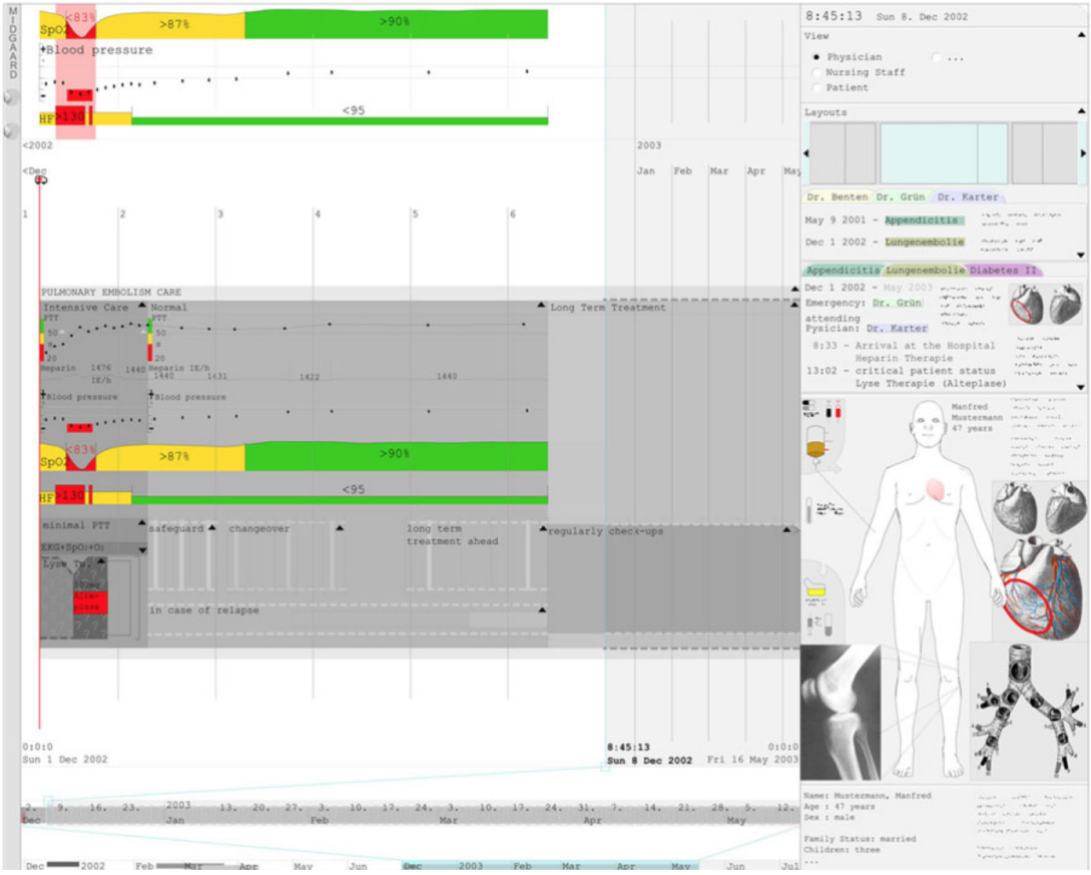

Figure 2.9: Example of Midgaard [8] interface with control panel and data visualizations.

When obtaining the data from Federal Almazov North-west Medical Research Centre (FANWMRC) medical research center[2] it became clear that there is a lot of noise including random data or backdated timestamps of records. This led to the emergence of a large number of implausible transfer sequences from physicians in charge of patients' treatment. Nodes [9] shows non-temporal force-directed layout of the resulting network considering all of its connections. In figure 2.10 we can see the visualization and an interactive simulation of an infection disease spreading in nosocomial contact network reconstructed from EMRs – (a) temporal force-directed layout with dynamic of patients' movement; (b) force-directed layout of entire time range of temporal network; (c) simulation of infection disease spreading in network through indirect contacts – purple edges; (d) search of possible ways of disease contamination for identified infected patient – red and yellow edges.

[2] http://www.almazovcentre.ru/?lang=en

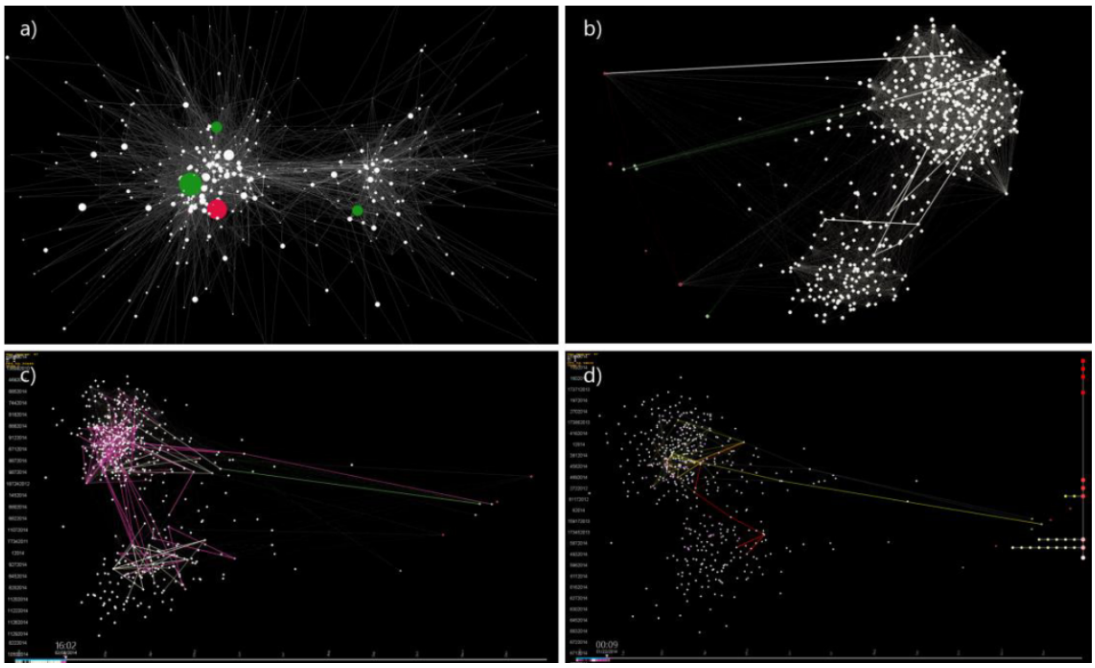

Figure 2.10: Example of NODES [9] visualization.

2.3 Multiple Patients

LifeLines2 [10] introduces new ways to visualize information and is an extension to Lifelines [4] that was designed to summarize the entirety of a single personal history record (e.g. a medical record). In contrast, LifeLines2 [10] displays selected subsets of the records from multiple patients. It consists of a visualization of records for temporal categorical data. For example figure 2.11 (a) shows individual records, and (b) shows the aggregation of events over time. (c) Contains controls to the basic operators: align, rank, and filter. (d) Shows controls for event distribution and grouping each triangle represents an event. The data is presented chronologically, and the records are ranked by the number of disease events. For instance, in Figure 2.11 it is easy to see the co-occurrence of pneumonia, influenza and asthma. However, in pneumonia and influenza incidences it is not clear, whether asthma occurred before or after. Users are forced to zoom in to each first occurrence of pneumonia and influenza for details, but each zoom can only reveal the details of a particular pneumonia and influenza event.

The distinguishing interaction technique of LifeLines2 [10] from the first LifeLines [4] is alignment. Users can align all records by a specific event type (for example, heart attacks). Every record's first heart attack event will then be aligned vertically. Records without heart attacks are filtered out. When the records are aligned, the timeline switches from an absolute calendar scale showing the current date of the events to a relative scale showing the amount of time before or after the date of the event used for alignment (i.e., one, two, or three days before/after). This allows users to spot trends in the timing of other events relative to the alignment event, in a group of patients. Users can also align by the nth

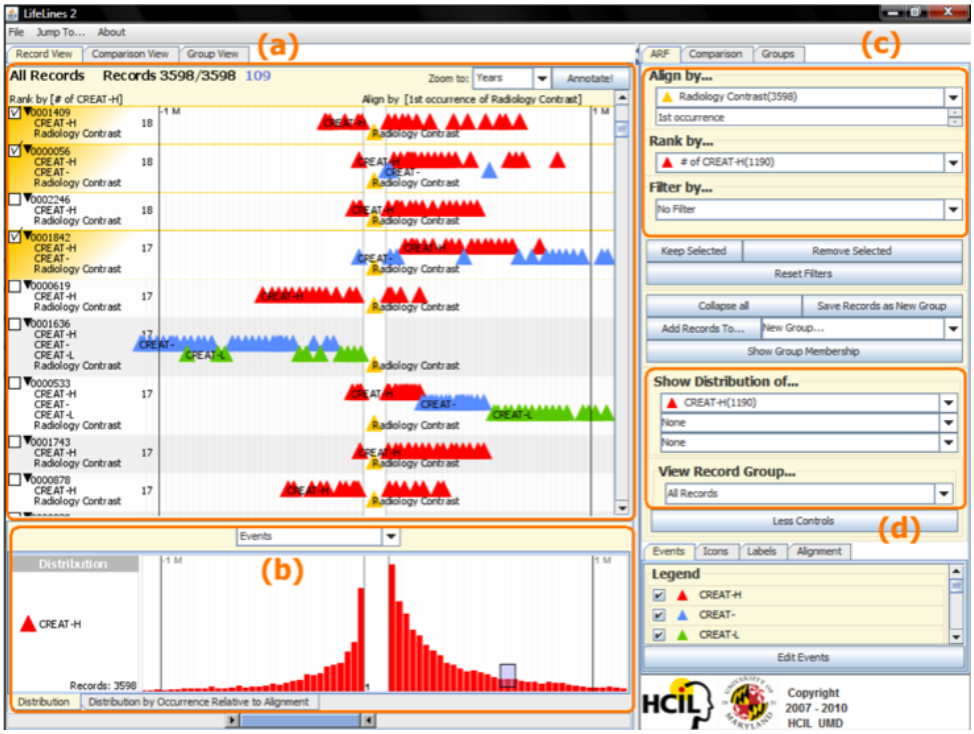

Figure 2.11: Example of LifeLines2 [10] interface.

occurrence of the event (e.g., the second heart attack), or they can align by all occurrences. In that case, the display of a record is duplicated for each occurrence of the event and each duplicate is aligned by one occurrence.

Similan [11] adopts the alignment concept from LifeLines2 [10] and allows users to pre-process datasets by aligning events by a sentinel event. Similan [11] displays all events in a timeline for each record. The extension to the rank-by-feature framework allows users to select a target record and then adjust the ranking criteria to explore the impact of result order.

Similan [11] will calculate a score that indicates how similar to the target record each record is and shows scores in the color coded grid on the left. The score color coding bars on the right show how the scores are color coded. Users then can sort the records according to these scores. The main panel also allows users to visually compare a target with a set of records. In this early prototype, the timeline is binned (by year, in this screen shot). If the users want to make a more detailed comparison, they can click on a record to show the relationship between that record and the target record in the comparison panel on the top. The plot panel at the bottom shows the distribution of records. To illustrate that, on figure 2.12 users can select categories of interest (left). The numbers of events in each category are displayed in the label. By clicking on the colored squares users can customize color. Users can choose to align events by selecting a sentinel category (middle). Weight for calculating total score can be adjusted using sliders and text boxes (right). Links in the comparison panel can be filtered using

these parameters.

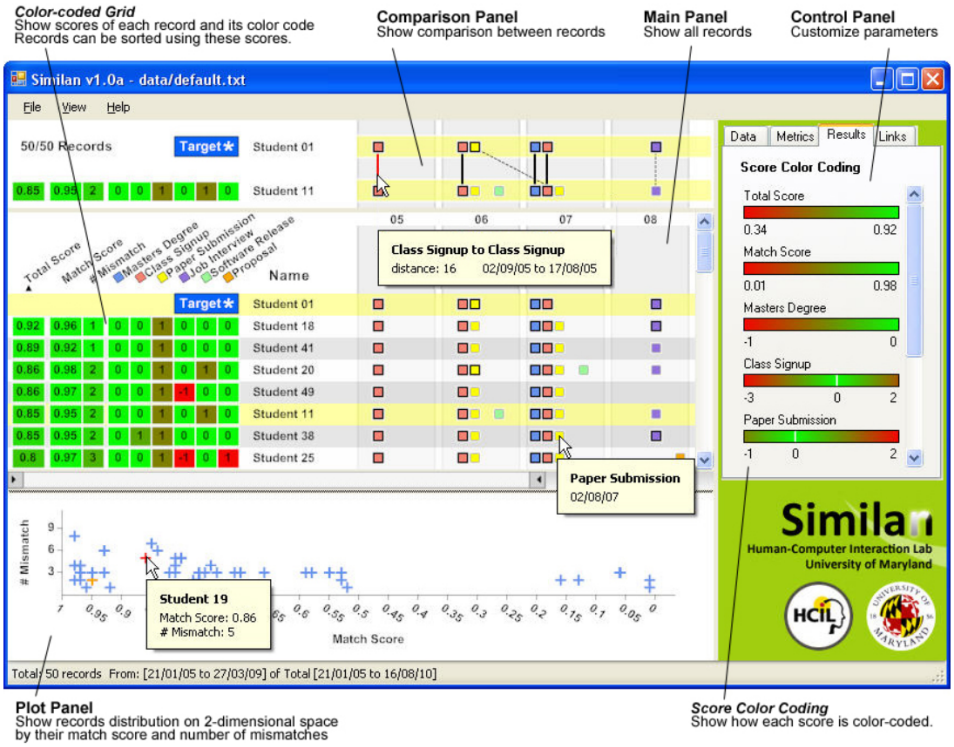

Figure 2.12: Similan [11] interface.

Focusing in the time variable we have VISITORS, VISualization of Time-Oriented RecordS [12], which combines intelligent temporal reasoning computational mechanisms with information visualization methods for display and exploration of time-oriented records of multiple patients. As depicted on figure 2.13 the two top panels display patients lists(denoted by A) and lists of time intervals (denoted by B), retrieved by computing the previous population-expressions that enables users to interactively specify temporal and knowledge-based constraints, through a graphical expression-specification module, which enables users to define the patient subsets selected for exploration. The graphs (denoted by C) show the data for a group of 58 patients with scatter plot combined with a line chart (graph 1) and for graph 2 and 3 are a multiple bar chart visualization.

In figure 2.14 the individual raw data is represented at a resolution level of seconds with respect to time granularity, but the population statistics are aggregated at a granularity of months, according to the user's current request. All laboratory test results for the specific cell count for patients treated during this period are displayed as (blue) X's (denoted by region 1). Through the density of the points, the user can judge the number of data instances belonging to each value or time period range. The top (red) line (denoted by 2) represents the monthly maximal value of the whole group. The tooltip provides detailed information about the maximal value), the ID of the patient with that value, and the observation time.

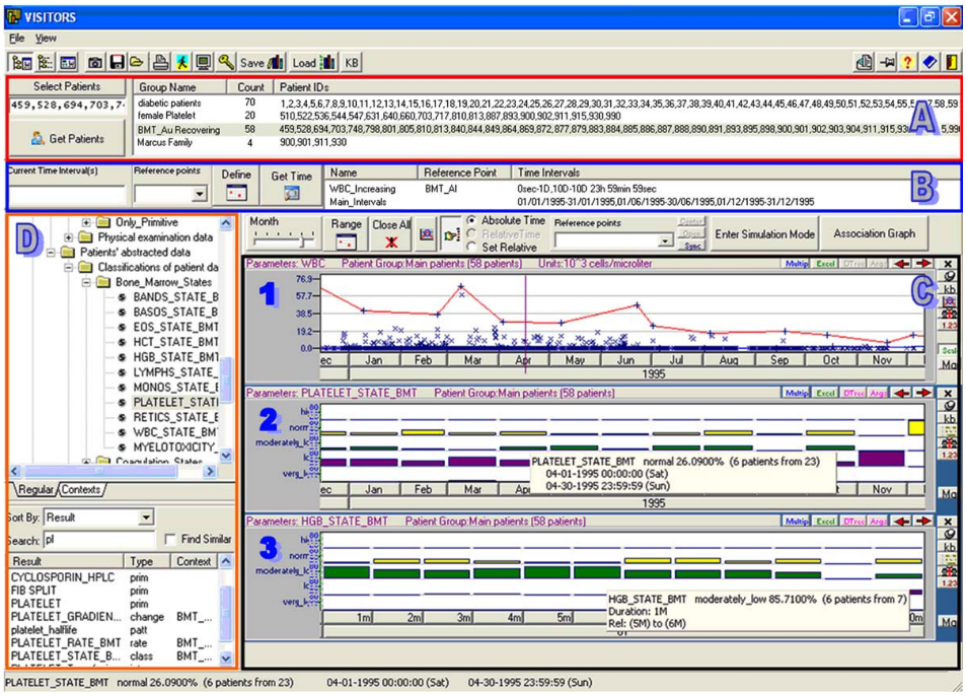

Figure 2.13: The VISITORS [12] main interface.

The bottom (blue) line (denoted by 4) and the middle (green) line (denoted by 3) represent the monthly minimal and mean values, respectively. On the left-hand side (denoted by 5) are displayed statistics for the whole time period. The three dotted lines drawn inside the panel indicate the mean value standard deviation for this period.

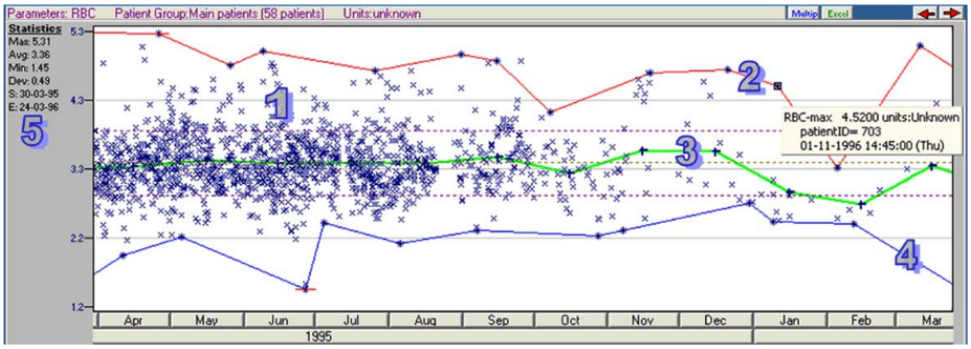

Figure 2.14: Data visualization for a specific cell count, in a raw concept, for a group of 58 patients [12].

With a resembling user interface, we have KNAVE-II [13]. Interpretation and exploration of longitudinal clinical data are a major part of diagnosis, therapy, quality assessment, and clinical research, particularly for chronic patients. KNAVE-II [13] is an intelligent interface to a distributed architecture specific to the tasks of query, knowledge-based interpretation, summarization, visualization, interactive exploration of large numbers of distributed time-oriented clinical data and dynamic sensitivity analysis of

these data. For example in figure 2.15 on the left hand side, a browser to the clinical domain's ontology from the ontology knowledge base, is shown. Operators represented as icons, in each panel, enable the user to perform actions such as: (a) The time-synchronization function [pin-shaped] icon, enables to synchronize the display of the panels according to the specified time in the selected panel (b) query the knowledge used to derive the concept through the "kb" icon, (c) the right and left direction arrows enable to skip to the nearest period (at the selected direction) data was found.

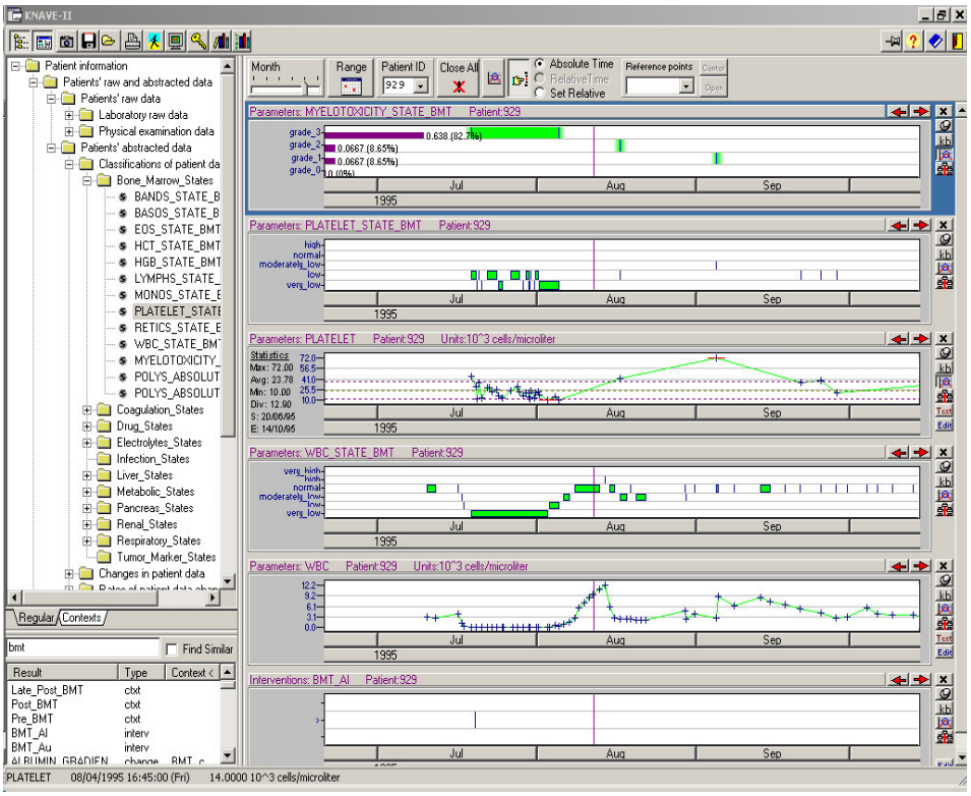

Figure 2.15: A view of an individual patient's data in the KNAVE-II [13] prototype.

In recent years, visualization methods have been developed to support both, the logical structure and the time-oriented aspects of computer-executable clinical treatment plans. However, visualizing the effects of applying treatment plans as well as supporting the exploration of effects on patients' conditions are still largely unresolved tasks. To fill this gap, CareCruiser [14] was developed to enhance known visualization methods to communicate the processes of treatment plan application together with their effects on a patient condition in an easily understandable way. To better understand, in figure 2.16 the logical view (a) communicates the logical structure of treatment plan execution by means of a flowchart-like representation. The lower left part (b) displays a tree graph to visualize the hierarchical structure of treatment plans and sub-plans; the time-oriented view (c) focuses on the temporal-qualities of applied treatment plans, clinical actions, and patient parameters. The time-oriented view is extended with step-wise interactive means to explore the effects of applied treatment plans on the patient's condition. This

screen shot shows one treatment plan that has been applied on three different patients (aligned vertically for comparison). The charts and treatment plans are colored according to the color scheme of the parameter values' distance to the intended value. Selecting ranges with big distance to the intended value with the range slider draws attention to critical cases and brings out the differences between the conditions of the three patients.

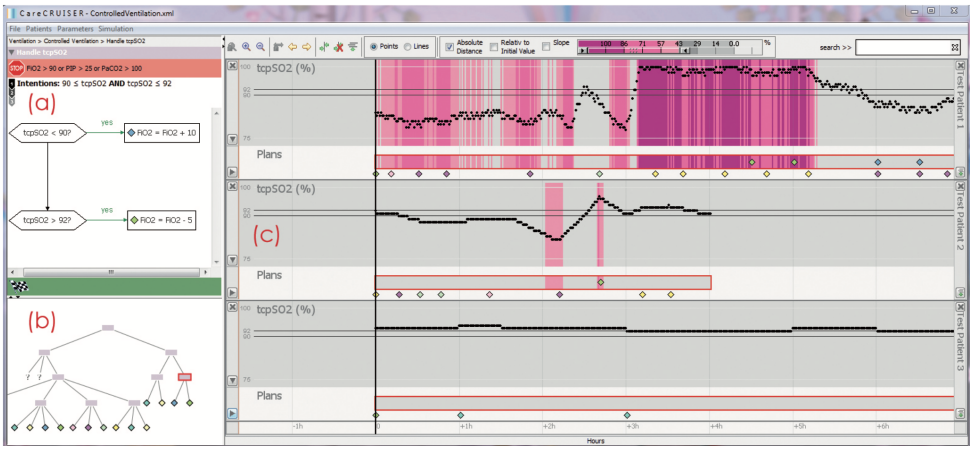

Figure 2.16: Example of CareCruiser [14] interface.

Innovating with uncommon visualizations, we have Gravi [15], VIE-VISU [16] and LINKVIS [17]. For Gravi [15] people are inquired and according to the answer a patient gave to a question, the patient's icon is attracted by the question's icon. The icons and visual elements can be moved, deleted, highlighted and emphasized by the user. Each change leads to an instant update of the visualization. In figure 2.17 two clusters show similar answering behavior of patients with positive therapy outcome (green icons) and those with negative outcome (red icons). The Star Glyphs (a connected line of the answer scores on each questionnaire for every patient) communicate exact values of all patients on showed questionnaires.

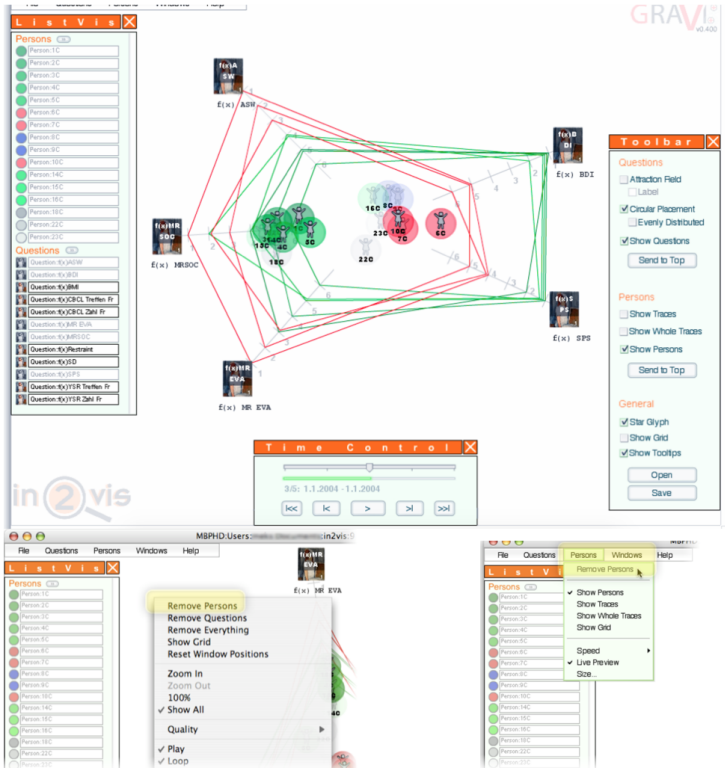

Figure 2.17: Example of Gravi [15] interface with control panel and data visualizations.

VIE-VISU [16] is a graphic visualization system witch uses metaphor graphics to represent a patient's status in neonatal intensive care. Graphics representation offers a wide variety of methods to support humans in the integration of data. VIE-VISU encodes categorical variables with geometrical shapes and their color, and encodes numerical variables by the size of glyph elements. The VIE-VISU display is a client-server application. It is embedded into the standard network of bedside and nursing workstations. The client offers a set of buttons for display management: it is possible to change the display between metaphor graphics and data values, moving around on time axis, changing time scales and select different patients to view. In figure 2.18 we have an example of a used shape. The value of a parameter is either depicted by the size of an object or by its color. The larger the value the larger the object, or darker the color. (Different visualizations, i.e types of ventilation, for the same patient).

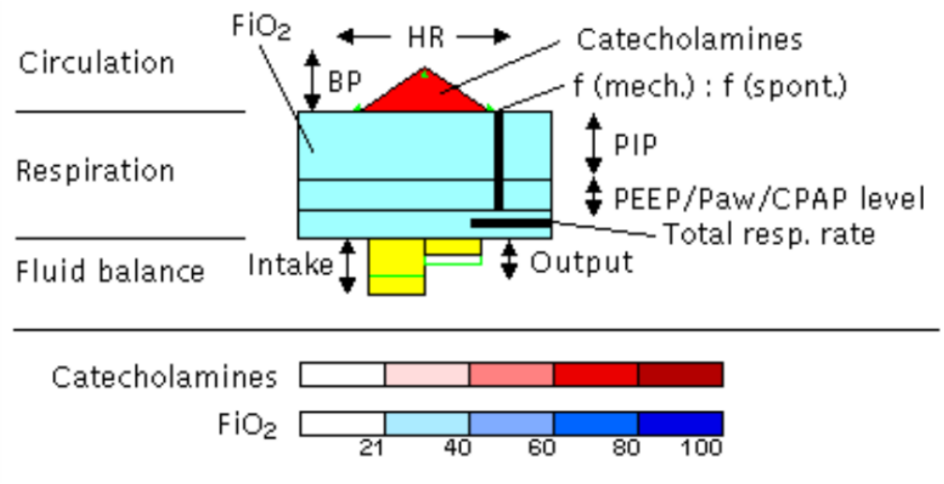

Figure 2.18: Example of VIE-VISU [16] geometrical shape.

Linkvis [17] is the prototype of an interactive exploration tool that supports the user by the analysis of heterogeneous and complex data. It uses multiple views to explore the same data in different perspectives, applies linking and brushing of the different views, and uses a hyperbolic hypothesis tree to guide users during the exploration process. Linkvis is based on three different types of Information Visualization techniques in which two selected techniques can be combined at a time (multiple views). As an example, on figure 2.19 in (a) Chernoff faces can simultaneously represent up to 10 dimensions, e.g. by varying shape of eyes and eyebrows, size and form of the mouth, and so forth. We only used four dimensions in our testing. The advantage of the technique lies in displaying single values and an overview at the same time, the disadvantage in communicating moods which may lead to undesirable associations and interpretations. In (b) two dimensional scatter plots represent the relationship between two variables. In contrast to Chernoff faces, numeric values can directly be read off the graphical representation. We extended the basic functionality of scatter plots to be able to represent the temporal directions of the parameters' changes. To illustrate these temporal changes, they added arrows to the value pairs. In (c) data are also displayed on vertical parallel co-ordinates. Data are represented as color coded dots interconnected by lines, one measurement per time point. It is possible to combine up to 10 variables within one graph.

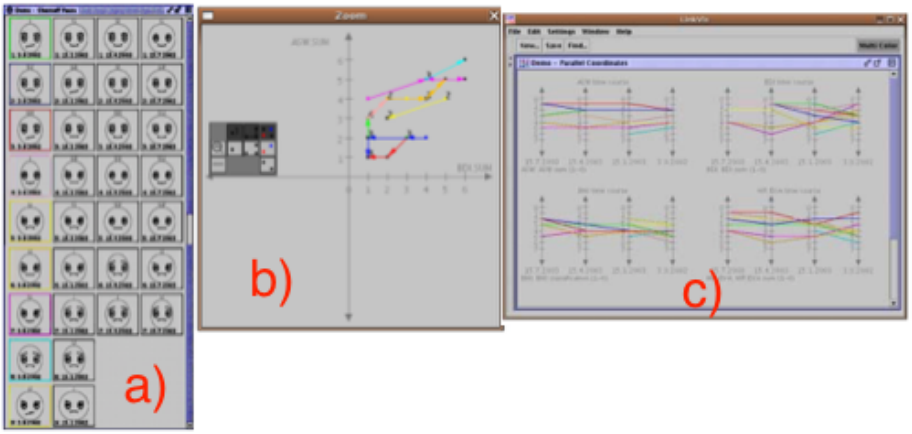

Figure 2.19: Linkvis [17] (a) Chernoff, (b) Two dimensional scatter plots, (c) Data displayed on vertical parallel co-ordinates.

2.4 Discussion

To compare and discuss all software, we have clustered software into three groups by the number of patients: population level, single patient and multiple patient. In general inside clusters their is a high percentage of matching features. We can conclude that the number of patients visualizing medical data is an important aspect when creating a visualization. They tend to have the same usability requirements making them similar within their cluster.

The cluster that gives users more features is the one who allow multiple patient comparison. They have more alternative ways to display information which helps every user to personalize his view as intended. LifeLines2 [10] and VISITORS [12] are the best example of it being two of the most complete works according to our study in table 2.1. The biggest shortcoming is the lack of tools to allow users to create their own visualization creating new derivative measures and code attributes. Those features make users more integrated with the system having a better system/user interaction.

A feature that also is not very common in the studied software, is the ”explore” option. Which means that user are not able to search for information in unknown location in the previous softwares database. It can restrict some research.

	Patient Comparison	General Trends	Detail on Demand	Pan+ Zoom	Highlight Relations	Code Attributes	Focus+ Context	Explore	Derive
DDC	NA	✓	✓	-	-	-	-	✓	-
AID	NA	✓	✓	-	✓	-	-	✓	✓
RODS	NA	✓	✓	-	-	-	-	✓	-
LifeLines	-	✓	✓	✓	✓	✓	-	-	-
VisuExplore	-	✓	✓	✓	✓	✓	-	-	✓
Caregiver	-	✓	✓	✓	✓	-	✓	-	-
MIVA	-	✓	✓	-	-	-	-	-	-
Midgaard	-	✓	✓	✓	✓	✓	✓	-	-
NODES	-	✓	✓	✓	-	-	-	✓	-
LifeLines2	✓	✓	✓	✓	✓	✓	-	-	-
Similan	✓	✓	✓	-	✓	✓	-	-	-
VISITORS	✓	✓	✓	✓	✓	-	-	✓	-
KNAVE-II	✓	✓	✓	✓	✓	-	-	-	-
CareCruiser	✓	✓	✓	✓	✓	-	✓	-	-
GRAVI	✓	✓	✓	-	✓	✓	-	✓	-
VIE-VISU	✓	✓	✓	-	-	-	-	-	-
Linkvis	✓	-	-	-	-	✓	-	-	-

Table 2.1: Software Comparison

3

Dashboard Alert

Contents

The final Solution is a visualization dashboard, that presents the patient history data, allowing users to find relevant patterns and timely alerts of problematic situations so doctors can act fast and adequately. It consist of a web application, running in any browser. Patient data were provided by by a research work from an other colleague [20].

3.1 Architecture

As we can see in figure 3.1, the system architecture is as follows: first it contains a database linked to the application server. That server communicate with the web server that provide our web application to the internet. The user browser present the web application in the user's personal computer.

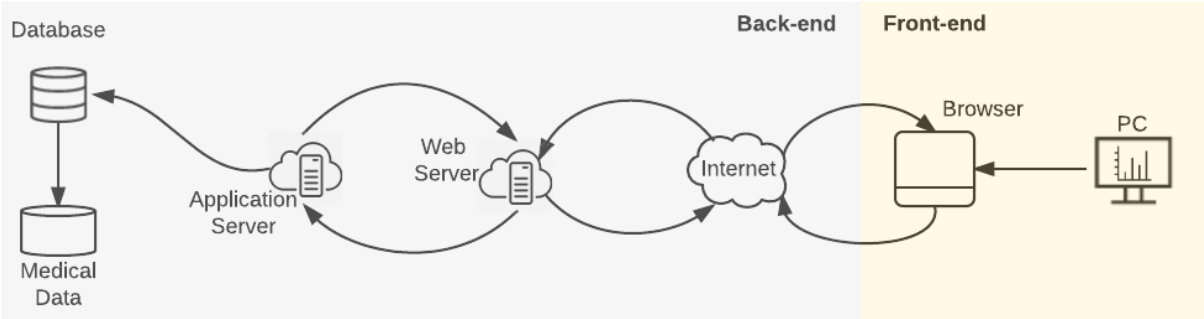

Figure 3.1: Proposed Architecture

3.2 Usability

To begin the implementation we started by determining some important concepts that will help get a better interface and interaction. There is a high number of people interested in the development of a system which can be affected positively or negatively by its introduction. So with the process below we will known who are they and what can we do.

3.2.1 Type of Users by *EASON*

According to Eason [21], there are 3 types of users: **primary**, **secondary** and **tertiary**. Primary are those that interact directly with the system. Those who seldom interact with the system or interact with the system through other people. Finally everyone else that is affected by the system. For our dashboard, primary users are dentist and dentist assistants, secondary users are patients and tertiary users are patients family and the hospital CEO. We used this approach because this method, with those three layers of user interaction with the system, allow us to classify quickly and accurately our users.

3.2.2 Stakeholders by *Sharp and colleagues*

The stakeholders by Sharp and colleagues [22] process, focus on stakeholders interaction: Baseline, Satellite, Suppliers, and Clients. Involving stakeholders can build trust, which can ultimately lead to increased consensus for your project or final decision. It can also increase transparency and lead to better decision making. With this approach, the interaction of our users are well defined making easier to understand the requirements of our solution and their priorities. So for this subsection we have:

End-users

- Baseline: Dentists;
- Client: Patient;
- Satellite: Dentist Assistant;

Developers

- Baseline: Ist Student and master's advisor;
- Supplier: IST - UL;
- Clients: Health care professionals;

Legislator

- Baseline: Health care professionals (dentists);

- Clients: Health care professionals;

Managers

- Baseline: Project Managers;
- Suppliers: Law executives, company funds and infrastructures;

3.2.3 Users profile & Focus group

We have three users profiles. Firstly we have a experienced dentist with more than five years of experience.

Profession: Dentists;

Experience: More than 5 years;

Personality: Impatient and persistent;

Secondly we have a young health care professional with less than five years of experience.

Profession: Cardiologist;

Experience: Less than 5 years;

Personality: Extrovert and sportive;

Lastly we have a young health care professional with less than five years of experience.

Profession: Oncologist;

Experience: Less than 5 years;

Personality: Active and generous;

Our focus group are health care professionals with active practice that like technology and want to be a part of a group that want to make a change in the way health care and technology interact with each other.

3.2.4 Personas & Scenarios

Personas communicate context in which the product will be used, behaviors and attitudes of users wants, needs or expectations from the product we are designing. They also show difficulties users want to overcome and goals that users may have. Personas will help better connection with users needs, as source of inspiration and keep things grounded in user research. Also in decision-making by validating decisions.

Scenarios are short stories that will answer several questions to set a context of the scenario: ”Where is the persona?”, ”When does it happen?” and ”Who else is around?”. Then we need a motivation for

what happens and user goal. With a motive, the user will act accordingly with the current situation: ”What happens?” ”How/why does it happen?”. Finally the consequences of those actions: ”What is the result?” ”Was the goal accomplished?”.

We have three different personas with the respective short biography and scenario.

Bran Smith
28 years old
Cupertino, CA
Dentist

Bio

Bran is a recent graduate having a Medicine degree from Birmingham University. He is working now for three years. In those three years he has tried to expand his knowledge and get as much experience as he can get. He started small with few patients but he is increasing his numbers trough time. Having such amount of work doesn’t mean that Bran don’t have also leisure moments. He likes to play online games with his friends and hangout at pubs.

Scenario

Bran is waiting for his first patient of the day and opens the patient file. He notices that he missed the last 4 appointments. Then he opens the Dashboard Alert to be aware how bad is to miss appointments so he can have a more informed talk with the patient. So he looks at the Heat Map and sees that people who miss between 4-6 times have much higher risk to contract a disease. When the patient arrives, Bran explains the risk for not coming more often. During the examination of the patient he spotted enlarged ganglia. So to be sure of the most effective treatment, he goes to the Dashboard Alert and with the Node Visualization looks for diseases that could be related as well as medication to see which path of treatment is more adequate.

Caitlin Allen
40 years old
Brooklyn, NY
Dentist

Bio

Caitlin is in the dentistry industry for about fourteen years. She started to gain experience by making some internships when she finished her course and working at a few great heath care facilities. Now she owns a clinic and wants quickly to make the clinic one of the best. In her spare time she practices basketball. She loves the sport and is huge fan of the Brooklyn Nets. Sport to her is a way for treating

her lack of patience and the stress accumulated from it.

Scenario

Caitlin is in her clinic and ask her assistant to check if the next patient was arrived. After waiting five minutes, she starts to be a little impatient and stressed out because it will delay all the other appointments. When he finally arrives, the patient starts do describe an uncommon symptoms. So for a better treatment, Caitlin opens Dashboard Alert and using the filter feature, highlights the possible diseases based on the described symptoms and rapidly builds a treatment plan by analyzing the Dashboard Alert information.

Luke Dunphy
31 years old
Brooklyn, NY
Dentist Assistance

Bio

Luke is health care professional for a long time now. He is very effective in what he does, he is always prepared for the needs of the patient. He makes sure that the patient gets what he needs to be more comfortable. He is seeking to be better everyday. He loves to do his morning run and eat vegetarian food. On holidays he volunteers in order to help homeless people.

Scenario

During an appointment while doing the patient check up, he spotted some anomalies and while he was diagnosing, Luke opened the Dashboard Alert and highlight diseases that have a primary diagnosis for what the patient might suffer. Between the three most probable diseases, gingivitis was the most likely. He processed with the right treatment, making the patient healthier.

3.2.5 Requirements

We have two types of requirements: functional that defines what the product must do and non-functional that constraints to the product and/or its development.

3.2.5.A Functional

Functional requirement is what the product must do support tasks. We have:

- Compare medicine X effect in patients that have Y disease.
- Correlate Y habit to X disease.
  - Find most common symptoms of X disease.
  - Find for symptoms Y and Z of the associated disease.
  - Find patients with X disease that have Y and Z symptoms.
- Compare in different periods of time the occurrence of X disease.
- Correlate number of medical appointments with number of diagnosed disease.
- Correlate patient age with disease X.
- Derive average age and sum total number of diagnosed diseases and correlate them.

3.2.5.B Data

Data requirements are the characterizations of data produced (output) and consumed (input), the type, volatility, persistency, volume and update. We have:

- Data input must be a JSON file.
- Data must be persistent.
- Data must be updated every month.
- Patients profile contains: id, name, biomedical attributes, diseases, medications, hygiene habits frequencies and general habits.

3.2.5.C Environment

Environment requirements are conditions under which the system operates. It can be physical e.g. lighting, noise, dust, heat and humidity. It can also be social where resource sharing (files, equipment)

are evaluated. Requirements also can be organizational where infrastructures, organization culture and structure are considered.

We have:

- Computer with Ethernet connection.
- The user must have a browser. It can be any available browser (Example: Mozilla Firefox[1]).

3.2.5.D User

User requirements are abilities and skills, difficulties and experience of the end user. We have:

- Must be dentist or dentist assistant in order to understand the information that the system provides.
- Must be fully capable of handling a computer (vision and physical).
- Minimum skill working with OSX or windows operative system.

3.2.5.E Usability

Usability requirements

- All data to be loaded in less than 3 seconds.
- Any task should be completed in less than 1 minute.
- 99% of the users should be able to finish the task.

3.2.6 Low Fidelity Prototype

In a first approach we developed the Low Fidelity Prototype on figure 3.2, where we have a chord diagram to correlate data between diseases and medication, a multiple bar chart to compare data in different periods of time and a line chart to see trough time the regularity of the attended appointment.

[1] https://www.mozilla.org/pt-PT/firefox/

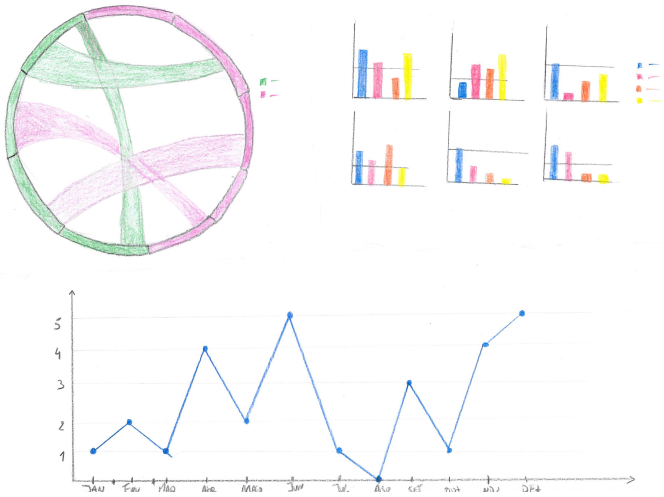

Figure 3.2: First Low Fidelity Prototype

Due to a change of data, some of the previous visualizations were adjusted not only in idiom but also in parameters. So, to have a better and clear visualization for the new data where we have diseases, medications and habits we changed some of the visualizations. With this new data, as we can see on figure 3.3, a new Low Fidelity Prototype was made. The chord diagram was kept because no data was changed. The coordinated view became less adequate so we replace it with a force oriented visualization that will work as a support to the chord diagram so that information be more clear with the proximity between nodes and clusters that may form. Related with patients appointments we removed the line chart and added a heat map. The heat map is shown by disease.

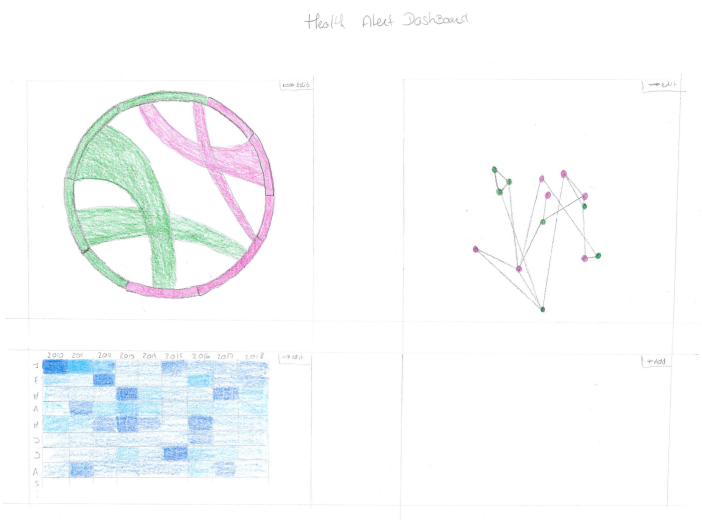

Figure 3.3: Second Low Fidelity Prototype

3.3 Implementation

3.3.1 Data

The data used in this project is a set of multiple JSON files. As we can see on figure 3.4 we have available in the first JSON the name, age, weight, height, body mass index (BMI), all the diseases that the patient had with start and end date followed by the prescribed medication. The medication frequency and dosage is also recorded. Finally hygiene habits, hygiene frequencies of those habits and other habits (non hygienic) are in the medical record.

```
1 {
2         "id": "85319fe0-fcde-11e6-b6cd-d93829301ef7",
3         "name": "Bertha Wenz",
4         "biomedicalAttributes": {
5             "age": 92,
6             "ageGroup": "86-95",
7             "weight": 68,
8             "height": 49,
9             "BMI": 0.010596885813148788
10        },
11        "diseases": [
12            {
13                "name": "Osteoporose",
14                "startDate": "2014-02-09T20:57:19.967Z",
15                "endDate": "2015-05-06T17:57:00.119Z"
16            }
17        ],
18        "medications": [
19            {
20                "name": "Broncodilatador",
21                "startDate": "2013-11-27T01:47:44.001Z",
22                "endDate": "2015-12-06T17:10:03.703Z",
23                "expectedFrequency": "Anual",
24                "recordedFrequency": [
25                    "2014-11-27T01:47:44.001Z"
26                ],
27                "dosage": [
28                    [
29                        {
30                            "name": "Broncodilatador",
31                            "dosage": 2
32                        }
33                    ]
34                ]
35            },
36            {
37                "name": "Esteróides",
38                "startDate": "2014-10-11T02:18:33.088Z",
39                "endDate": "2017-01-07T09:51:25.428Z",
40                "expectedFrequency": "Mensal",
41                "recordedFrequency": [
42                    "2014-10-11T02:18:33.088Z",
43                    "2015-01-09T03:18:33.088Z",
44                    "2015-02-08T03:18:33.088Z",
45                    "2015-03-10T03:18:33.088Z",
46                ],
47                "dosage": [
48
49                    [
50                        {
51                            "name": "Esteróides",
52                            "dosage": 3
53                        }
54                    ],
55                    [
56                        {
57                            "name": "Esteróides",
58                            "dosage": 3
59                        }
60                    ]
61                ]
62            }
63        ],
64        "habits": [],
65        "habitsHigiene": [],
66        "habitsGeneral": [],
67        "lastVisit": "2014-05-[illegible]T00:40:10.368Z",
68        "lastVisitPeriod": "Mais que 2 anos e menos que 5 anos"
69    }
```


Figure 3.4: Example of Patients JSON

3.3.2 Technology

To develop the dashboard, we searched for multiple technologies and after a extended research the most appropriate was the combination of D3.js[2] and React.js[3].

D3.js it's a JavaScript library for manipulating documents based on data. D3 helps you bring data to life using HTML, SVG, and CSS. D3's emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation.

In the other hand React.js is a JavaScript library for building user interfaces. It allows developers to create large web-applications that use data and can change over time without reloading the page. It aims primarily at providing speed, simplicity, and scalability. React processes only user interfaces in applications. This corresponds to the View in the Model-View-Controller (MVC) pattern, and can be used in combination with other JavaScript libraries or frameworks in MVC, such as AngularJS[4].

To use both, we had to make some adaptations in the source code of the visualization in D3. Those modifications allowed React to recognize all parameters and data that the D3 files have.

3.3.3 Visualizations

3.3.3.A Chord Diagram

One of our tasks was to correlate medications and diseases. For that we believe that the chord diagram was more suitable. The JSON (figure 3.5) used derived from the original but now with only what we need: the sum of the number of times that diseases and medications appear together.

[2] https://d3js.org/
[3] https://reactjs.org/
[4] https://angularjs.org/

```
[
    {
      "who": "Análgésicos",
      "overlap": "Anti-depressor",
      "incidences": 1
    },
    {
      "who": "Glaucoma",
      "overlap": "Aspirina",
      "incidences": 1
    },
    {
      "who": "Osteoporose",
      "overlap": "Anti-diabéticos",
      "incidences": 1
    }
]
```

Figure 3.5: Example of the chord diagram JSON

Firstly we build this chord diagram (figure 3.6) where we had two different halves: in one side we had disease and in the other side medication. Each chord when a element was selected it collapsed into that element showing only the other chord elements that were connected to it. When moving the mouse away the chord restores its original form.

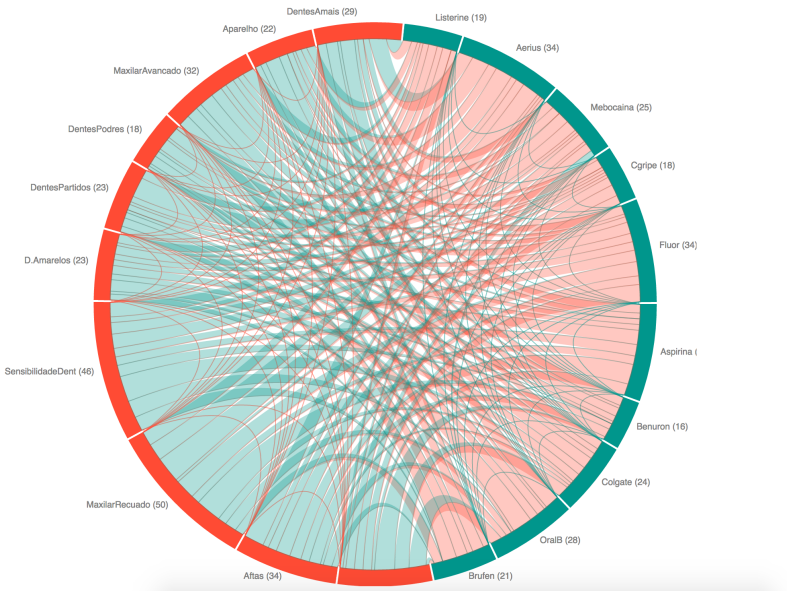

Figure 3.6: First attempt of chord diagram

However we wanted to allow users to filter the visualization permanently so we create this new version where on click, the selected element is removed from the chord and added to a list of removed elements. That list, contains the removed elements and can be added to the chord (figure 3.7) diagram again by selecting them on the list.

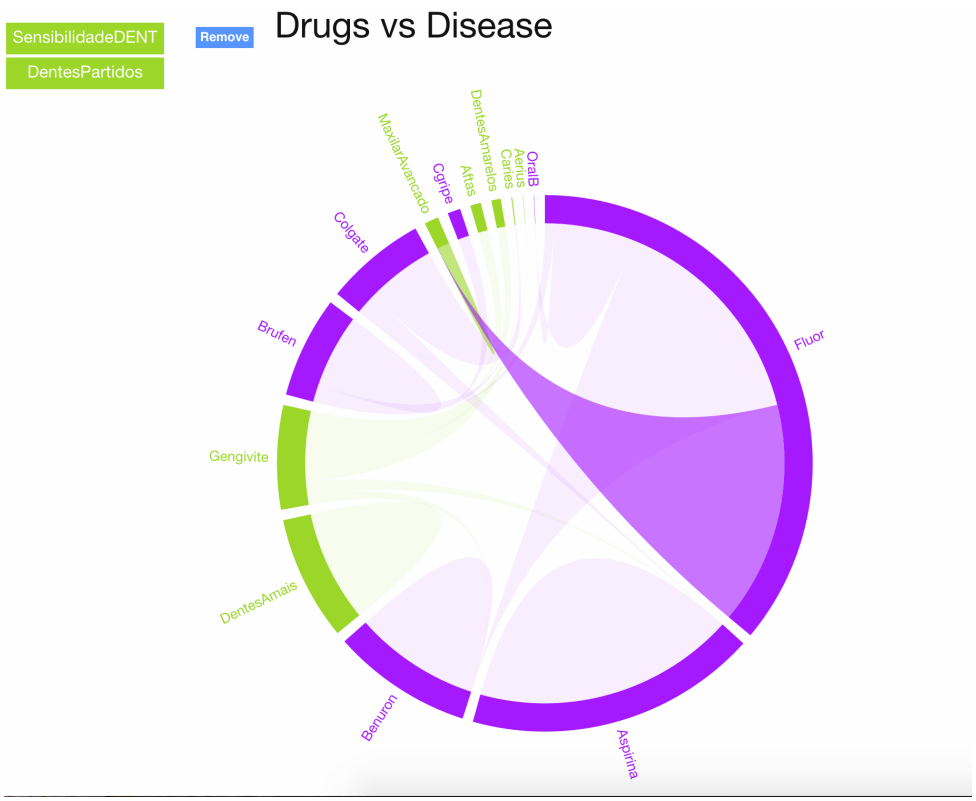

Figure 3.7: Second attempt of chord diagram

To make the integration with the dashboard, this visualization was not suitable because it was partially made with Angular.js[5]. So we were forced to remake the chord (figure 3.8) in order to have a smooth integration in the dashboard. This new chord diagram is now entirely in D3.js. Our first visualization is now finished.

[5]https://angularjs.org/

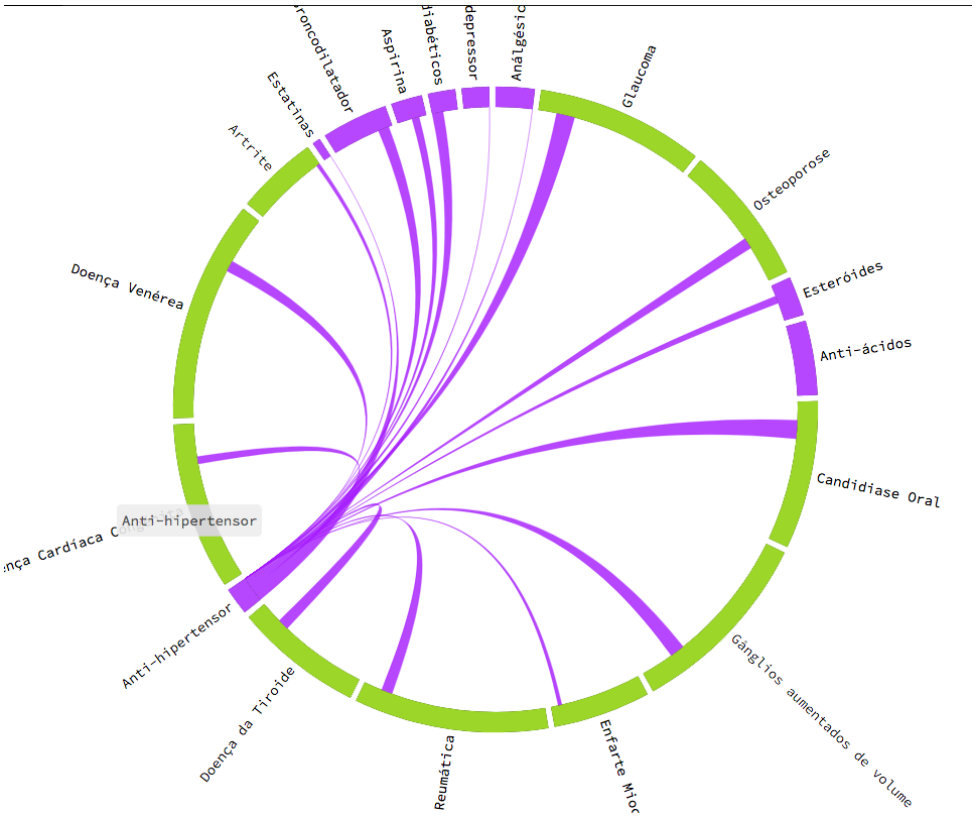

Figure 3.8: Final attempt of chord diagram

3.3.3.B Node Diagram

For the second visualization we implemented a node diagram in order to complement the chord diagram to correlate diseases and medications. For this visualization, the data, figure 3.9, is divided into two parts, and we identify groups, relations and their value.

```
1  {
2    "nodes": [
3      {
4        "id": "Análgésicos",
5        "group": 1
6      },
7      {
8        "id": "Anti-depressor",
9        "group": 1
10     },
11     {
12       "id": "Glaucoma",
13       "group": 2
14     }
15   ],
16   "links": [
17     {
18       "source": "Análgésicos",
19       "target": "Anti-depressor",
20       "value": 1
21     },
22     {
23       "source": "Glaucoma",
24       "target": "Aspirina",
25       "value": 1
26     },
27     {
28       "source": "Osteoporose",
29       "target": "Anti-diabéticos",
30       "value": 1
31     }
32   ]
33 }
```

Figure 3.9: Example of Node JOSN

With this new visualization (figure 3.10) users could now have a more clear view of the relations between diseases and medications. We can see clusters due the proximity of the nodes. Related nodes are closer to each other.

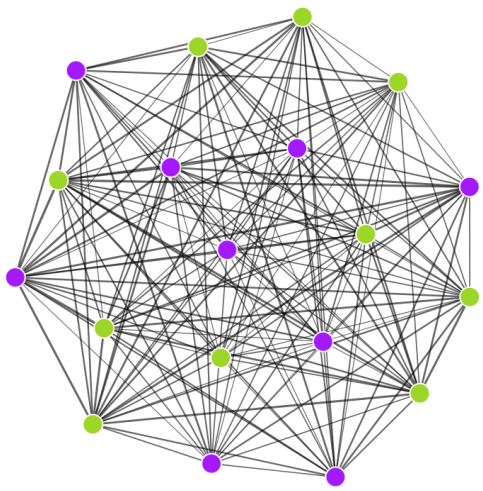

Figure 3.10: Force Diagram

3.3.3.C Heat Map

We made a heat map for the third visualization. The data is contained in a JSON (figure 3.11) with an array of objects with the month, year and value of incidence. This visualization (figure 3.12) will evidence the incidences of the different diseases through time. For each month we have the number of incidences from 2012 to 2017. We were now able to see if a disease was increasing its incidences or not.

```
1 [
2     {
3         "month": 1,
4         "year": 1,
5         "value": 0
6     },
7     {
8         "month": 1,
9         "year": 2,
10        "value": 0
11    },
12    {
13        "month": 1,
14        "year": 3,
15        "value": 33
16    },
17    {
18        "month": 1,
19        "year": 4,
20        "value": 0
21    },
22    {
23        "month": 1,
24        "year": 5,
25        "value": 3
26    }
27    ]
28
```


Figure 3.11: Example of Heat map JSON

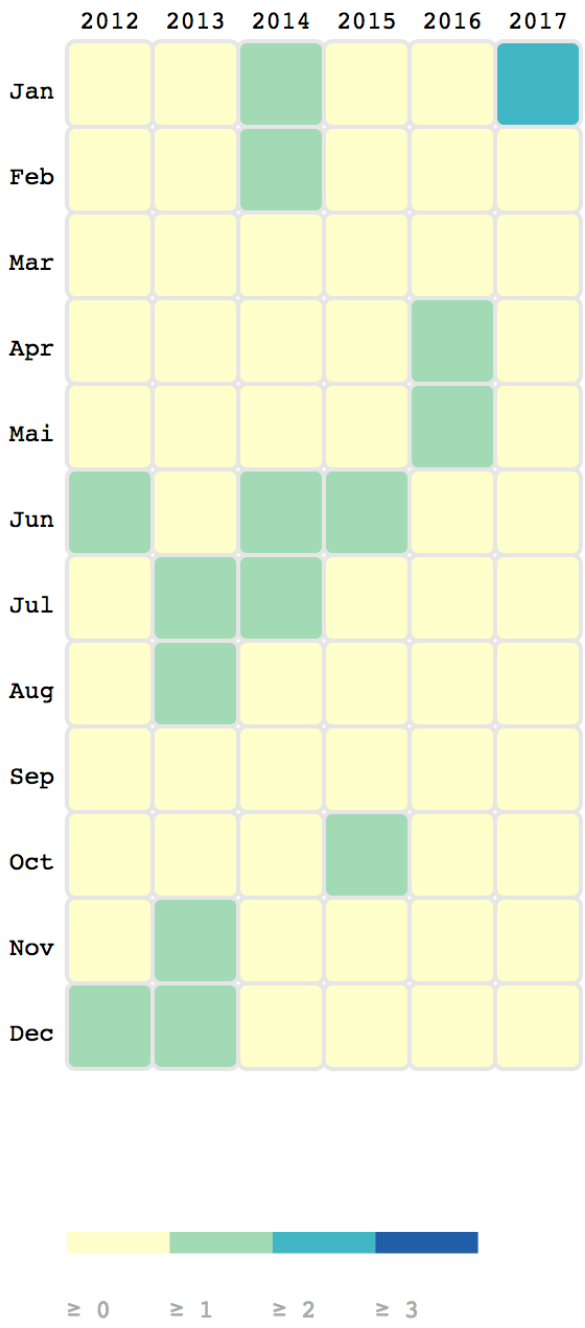

Figure 3.12: Heat map for 'Glaucoma'

3.3.4 Bar Chart

For the fourth visualization, we made a bar chart. The data is organized as an array of objects containing age and frequency (figure 3.13). The bar chart (figure 3.14) will allow to compare diseases

```
1  [
2      {
3          "age": "16-25",
4          "frequency": 5
5      },
6      {
7          "age": "26-35",
8          "frequency": 9
9      },
10     {
11         "age": "36-45",
12         "frequency": 7
13     },
14     {
15         "age": "46-55",
16         "frequency": 18
17     },
18     {
19         "age": "56-65",
20         "frequency": 0
21     },
22     {
23         "age": "66-75",
24         "frequency": 3
25     }
26     ]
27
28
```

Figure 3.13: Example of Bar Chart JSON

the number of incidences for the different age ranges. With that we can preemptively know what disease a person might have in the future and avoid it easily.

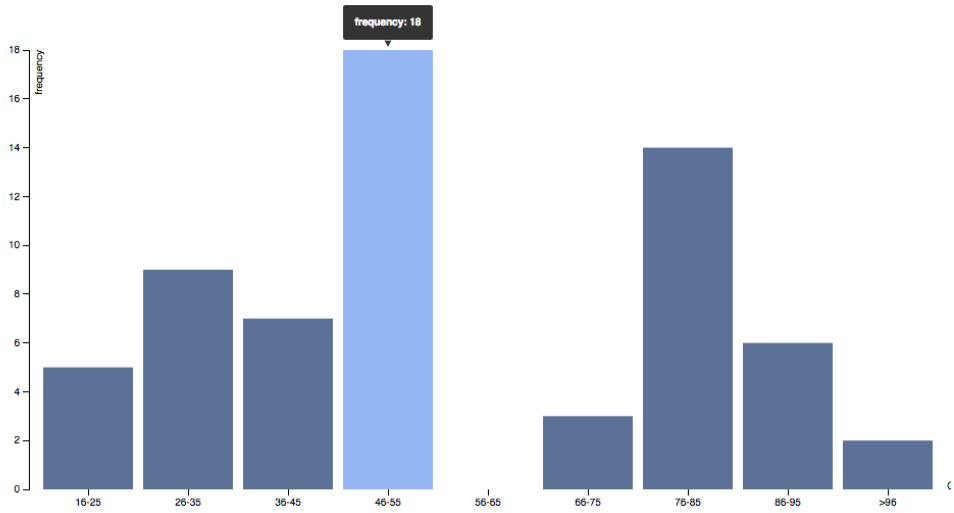

Figure 3.14: Bar chart for 'Glaucoma'

3.3.5 Integration with React

Both React and D3 are two excellent tools designed with goals that sometimes collide. Both take control of user interface elements, and they do so in different ways.

The D3 in D3.js stands for data-driven documents. D3.js is a low-level library that provides the building blocks necessary to create interactive visualizations. It uses web standards such as SVG, HTML, Canvas, and CSS to assemble a front-end toolbox with a vast API and almost limitless control over the look and behavior of visualizations. It also provides several mathematical functions that help users to calculate complex SVG paths. D3.js loads data and attaches it to the DOM. Then, it binds that data to DOM elements and transforms those elements, transitioning among states if necessary.

D3.js selections are similar to jQuery objects, because they help us deal with SVG complexity. The way this is done is comparable to the way jQuery deals with HTML DOM elements. Both libraries also share a similar chain-based API and the use of the DOM as data storage.

React is a JavaScript library that helps us build user interfaces by composing components. These components keep track of their state and pass their properties to re-render themselves effectively, optimizing the performance of the application. The virtual DOM, which is a representation of the DOM's current state, is the technology that enables React's re-rendering optimizations. The library uses a complex diff algorithm to understand which parts of the application need to re-render when the conditions change. This diff algorithm is called the "reconciliation algorithm."

Both React and D3 share the goal of helping users deal with the DOM and its complexity in a highly

optimized way. They also share a preference for pure functions — code that, for a given input, always returns the same output without incurring side effects — and stateless components.

However, the shared concern about the DOM makes these two opinionated libraries clash when determining which is to render and animate the user interface elements. We saw different ways to solve this dispute. We established a hard rule: They should never share DOM control.

The approach we followed was to give our D3 code as much DOM control as possible. It employs a React component to render an empty SVG element that works as the root element of our data visualization. Then it uses the *componentDidUpdate* lifecycle method to, using that root element, create the chart using the D3.js code we would employ in a vanilla JavaScript scenario.

When integrating React and D3.js, we can do so at different levels. The approach we followed is to give our D3 code as much DOM control as possible. It employs a React component to render an empty SVG element that works as the root element of our data visualization. Then it uses lifecycle methods to, using that root element, create the chart using the D3.js code we would employ in a vanilla JavaScript scenario.

This is a simple solution that works fine most of the time. It is also the most natural solution when you are using D3.js charts that were already working somewhere else.

On the downside, mixing both React code and D3.js code within a React component could be seen as a bit gross, incorporating too many dependencies and making that file too long to be considered quality code.

Om figure 3.15 and 3.16 we can see the resulting dashboard.

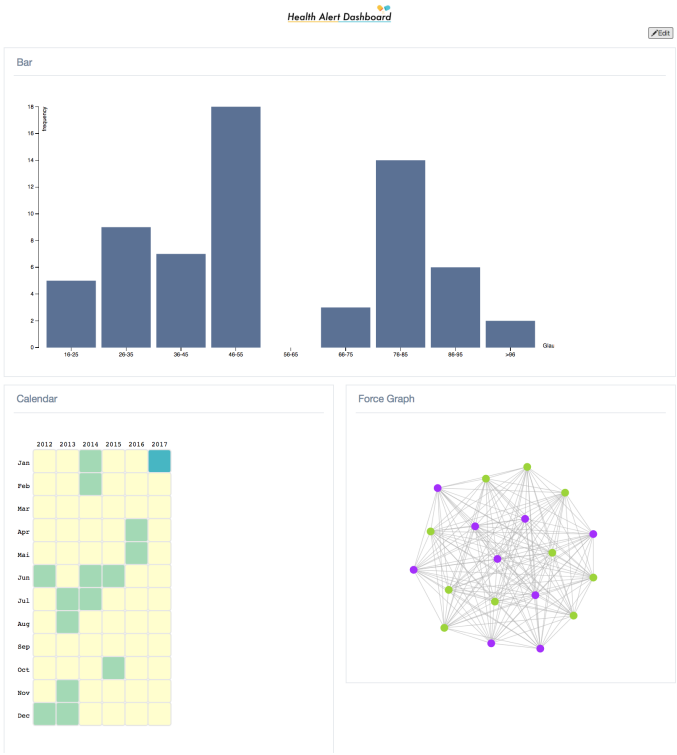

Figure 3.15: Dashboard Screenshot

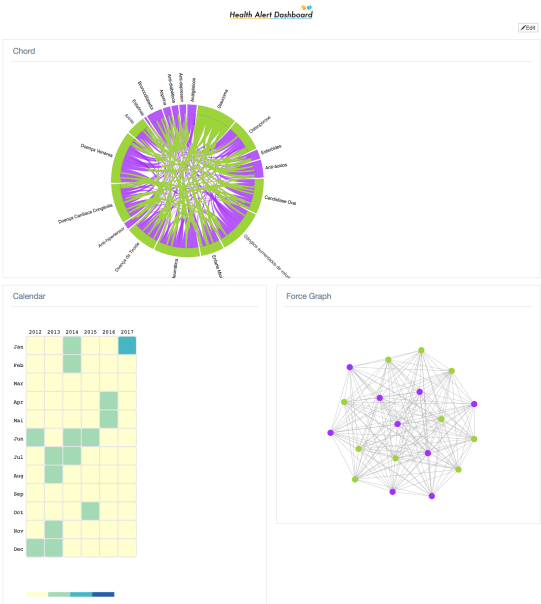

Figure 3.16: Other Dashboard Screenshot

4

Evaluation

Contents

We followed an iterative and incremental approach and it was necessary to test the solution at different phases of the development :

- We did heuristic evaluation with experts (Low-Fidelity Prototype). Low-fidelity prototypes are quicker to create. They allow to have a quick and easy tangible representation of the project concept for getting quick feedback. Based on our data set we created tasks and questions [18] that would explore most of the information we could retrieve from it. For that we used a specific set of heuristics for information visualization.

- Formative evaluation with users. We gathered three users and brainstormed new ideas with the help of two methods: *Card sorting*[1], *Think-aloud*[2] and *Six thinking hats*[3]. To perform *Card sorting*, we distributed cards with different type of topics (diseases, treatments etc.) and some blank cards where the users wrote their categories. After that, users ordered the cards within their categories. We obtained the perception of where concepts are clustered in users minds. Afterward, in *Six thinking hats*[4], we sat all together and talked about appointments and technology in medical appointments respecting the color rule of the hat. The session started with everyone assuming the Blue hat to discuss how the meeting was to be conducted and to develop the goals and objectives. After that we moved to white hat thinking to collect first facts so we could evolve from there. The discussion then moved to Red hat thinking in order to collect opinions and reactions to the problems of technology in medicine. This phase also allowed to develop constraints for the current solution such as who will be affected by the problem and/or the solution. Next the discussion moved to the Yellow and to the Green hat in order to generate ideas and possible solutions. Next, the discussion moved between White hat thinking as part of developing information and Black hat thinking to develop criticisms of the solution set. In the *Think-aloud*[5] session each user performed two tasks and while performing them, we tracked their satisfaction using the solution by hearing what they were thinking about every step of the task.

- Summative evaluation with users. When performing tasks, we monitored the users errors, time etc. With the result of those tests, we saw if we have accomplished our objectives: to have a efficient way to visualize medical data making it promptly available to users.

In the evaluations we had access to health care professionals from different areas that could validate and evaluate the prototypes.

[1] https://www.usability.gov/how-to-and-tools/methods/card-sorting.html
[2] https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/
[3] De Bono, Edward. Six thinking hats. London: Penguin, 1989.
[4] De Bono, Edward. Six thinking hats. London: Penguin, 1989.
[5] https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/

4.1 Usability Tests

4.1.1 Protocol

For usability tests, we invited 3 health care professional that, due availability issues, three of them haven't their medical specialization in dental care. We met at Biblioteca Palácio Galveias[6], Lisbon, and started with a small talk to make everyone more comfortable. In order to understand how they felt now about the evolution of technology in health care, their needs and worries.

4.1.2 Six Thinking Hats

We performed *Six Thinking Hats* [7] that is a simple, effective parallel thinking process that helps people be more focused and mindfully involved. To increase the engagement of participants we have made a slight change and renamed it *The six thinking syringes*.

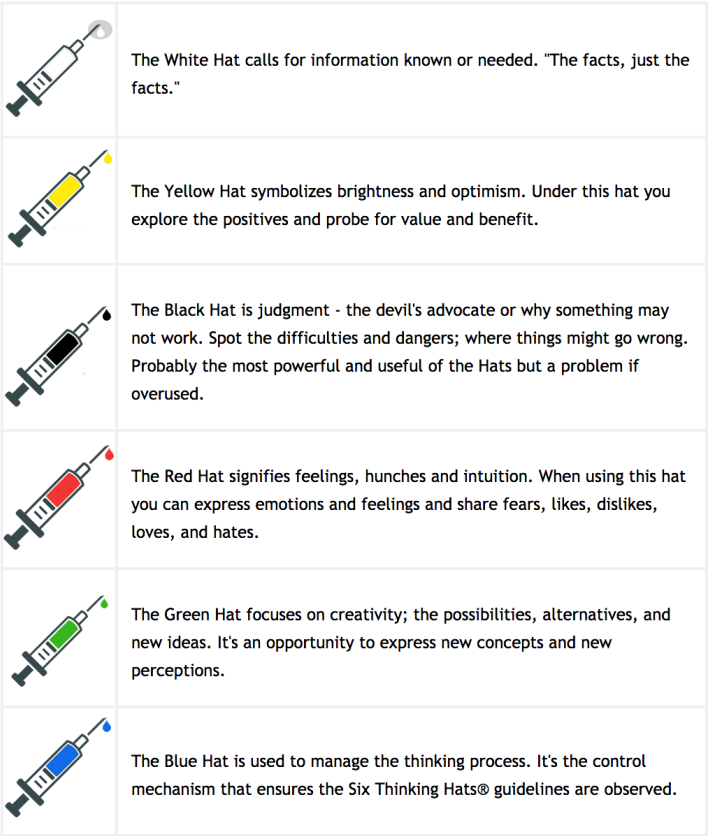

Figure 4.1: Six Thinking "Syringe"

[6] http://blx.cm-lisboa.pt/gca/index.php?id=404
[7] www.debonogroup.com

4.1.2.A Results

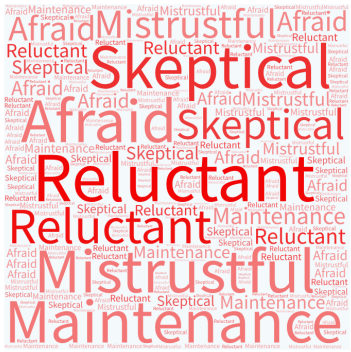

Figure 4.2: Word cloud that represent the results from red hat discussion.

Figure 4.3: Word cloud that represent the results from yellow hat discussion.

Figure 4.4: Word cloud that represent the results from black hat discussion.

4.1.2.B Discussion

From this process we learned more about our users and gathered user expectations. From the red hat, the participants showed that they are reluctant about abandoning their current way for searching this kind of patient information and some are not very confident of the accuracy of the data and if it can be generalized. In the other hand for the yellow hat we have optimism about the preventiveness of the solution, we retrieve information very quickly, easy to understand and for recent graduate can be very helpful like a mentor. Finally for the black hat it was mentioned a problem that what will happen if the dataset fields changes, who is responsible for the information that is represented and its consequences if applied to a patients. For the green hat we had just a suggestion, mobile version, to use in tablets in appointments.

4.1.3 Pictive

After, we made groups for *Pictive*. *Pictive*, Plastic Interface for Collaborative Technology Initiatives through Video Exploration, is a paper mock-up technique that allows users to participate in the development process. A *Pictive* session is a representation of a graphical user interface on paper. The ultimate goal of a PICTIVE is to simplify the design process enough that non-technical users are empowered to participate in it.

A *Pictive* is usually made from simple office supplies like pens, paper, Post-It stickers, and paper clips. We used these stationary to represent elements of the project, including visualization, menu bars, and special icons. During a design session, users manipulate the mock-up so it becomes easier for them to use.

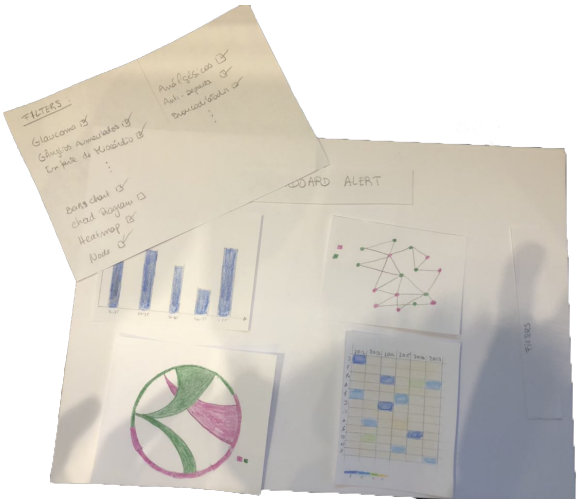

Figure 4.5: Initial components for the Pictive

4.1.3.A Results

Figure 4.6: Example from a participant

4.1.3.B Discussion

From the Pictive we got different suggestions of dashboard from our users and we didn't have consensus in the number of visualizations, position of those visualizations and the position of the filters menu. This is interesting to see that there isn't a one way to do the dashboard interface but is also negative because it shows that users don't have the same vision of the final product but have the similar needs, which can be confusing to the development process. Our dashboard will follow some of the users ideas and suggestion like not more than three chart to the main page and some will be discarded like the filter button not having no alignment in the web page.

4.1.4 (Open) Card Sorting

Participants were asked to organize topics that are provided into groups that make sense to them and then name each group they created in a way that they feel accurately describes the content. We used an open card sort to learn how users group content and the terms they give each category.

Figure 4.7: Initial cards from Card Sorting

4.1.4.A Results

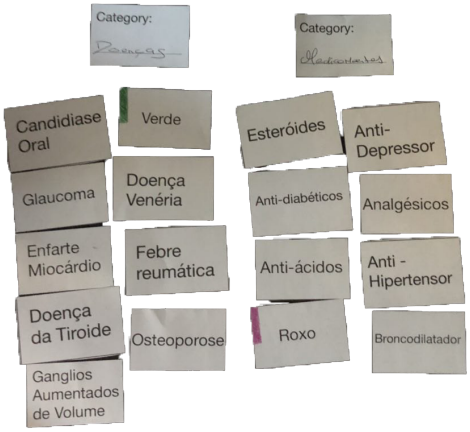

Figure 4.8: Example from a participant

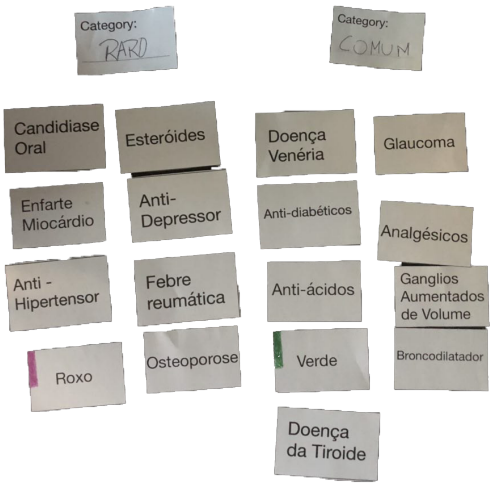

Figure 4.9: Other example from a participant

4.1.4.B Discussion

From the cart sorting we concluded that diseases and medication can be at completely different groups but can also be clustered in groups with common properties like rarity of diagnosing / prescribing, probabilities of simultaneous occurrences and usage. So our force diagram will confirm this by showing heterogeneous clusters.

5

Conclusion

Trough the years different algorithms and visualizations were created to help transforming big quantity of health data into information useful for experts in different fields (i.e LifeLines [4] for doctors in hospitals and RODS [3] for epidemic control).

Some of the solutions we have seen are in some way incomplete or have gaps and based on those works with the development of this dissertation we will study ways in which an effective visualization can fix some of those gaps providing an interactive dashboard to provide health care experts with relevant information at the right time in the right way.

During this work we learned about some of the research that have been made around information visualization and health care. With that knowledge, the challenge to create a dashboard were a very interesting one. We searched for similar approaches and started to build our own. We started with user research defining user types, personas and our focus group. From the workshop done, Six Thinking hats, Pictive and Card Sorting, we could conclude that users are still averse to change. They are already too familiar with their tools and don't want to learn everything from scratch again. In the other hand the dashboard does in fact give the user any information he/she wants, with the given data, very quickly and in a orderly way. The thing they most liked was that all the information is centralized in one single place. In the moment of creating the dashboard, we made the charts and the dashboard in different technologies. The toughest part was the integration of the D3.js charts into the React.js dashboard because of the many differences of handling the DOM.

6

Future work

The solution now it's in a initial state. In the future it can be upgraded with many features regarding new filters, to allow users to have a better and more focused view, more communication between charts to make the system more interactive and add new charts for more diverse and complete information. Interactivity is very important to user interaction, so it should be refined and perfect the best we can. Filtering by disease or by medication will be possible. A set of check boxes will be provided and when a disease or medication is selected, it will to switch its visibility on the active charts. Users are already able to choose the visualization they want to see. It requires about 3 step to do so. We can reduce the number of step by creating a interactive menu that with 1 step we can filter everything we want. Between visualizations, same concepts will be highlighted simultaneously. All these new features will increase significantly the quality of our Dashboard Alert.

Bibliography

[1] S. Rashid, “Disease data correlator,” 2008.

[2] S. Al-Hajj, I. Pike, and B. Fisher, “Interactive dashboards: Using visual analytics for knowledge transfer and decision support,” *Public Health’s Wicked Problems: Can InfoVis Save Lives? IEEE Visual Analytics Science and Technology, Information Visualization, Scientific Visualization*, 2013.

[3] S. Purushe, “Evaluation of user interfaces for medical data visualization: a critical study of ui of emergency medical infovis,” 2010.

[4] C. Plaisant, R. Mushlin, A. Snyder, J. Li, D. Heller, and B. Shneiderman, “Lifelines: using visualization to enhance navigation and analysis of patient records.” *Proceedings of the AMIA Symposium*, 1998.

[5] A. Rind, W. Aigner, S. Miksch, S. Wiltner, M. Pohl, T. Turic, and F. Drexler, “Visual exploration of time-oriented patient data for chronic diseases: Design study and evaluation,” *Symposium of the Austrian HCI and Usability Engineering Group*, 2011.

[6] D. Brodbeck, R. Gasser, M. Degen, S. Reichlin, and J. Luthiger, “Enabling large-scale telemedical disease management through interactive visualization,” *European Notes in Medical Informatics*, 2005.

[7] A. Faiola and C. Newlon, “Advancing critical care in the icu: a human-centered biomedical data visualization systems,” *International Conference on Ergonomics and Health Aspects of Work with Computers*, 2011.

[8] R. Bade, S. Schlechtweg, and S. Miksch, “Connecting time-oriented data and information to a coherent interactive visualization,” *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2004.

[9] A. Karsakov, A. Moiseev, K. Mukhina, I. Ankudinova, M. Ignatieva, E. Krotov, V. Karbovskii, S. V. Kovalchuk, and A. O. Konradi, “Toolbox for visual explorative analysis of complex temporal multiscale contact networks dynamics in healthcare,” *Procedia Computer Science*, 2016.

[10] T. D. Wang, C. Plaisant, A. J. Quinn, R. Stanchak, S. Murphy, and B. Shneiderman, "Aligning temporal data by sentinel events: discovering patterns in electronic health records," *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2008.

[11] K. Wongsuphasawat and B. Shneiderman, "Finding comparable temporal categorical records: A similarity measure with an interactive visualization," *Visual Analytics Science and Technology, 2009. VAST 2009. IEEE Symposium on*, 2009.

[12] D. Klimov, Y. Shahar, and M. Taieb-Maimon, "Intelligent visualization and exploration of time-oriented data of multiple patients," *Artificial intelligence in medicine*, 2010.

[13] Y. Shahar, D. Goren-Bar, M. Galperin, D. Boaz, and G. Tahan, "Knave-ii: A distributed architecture for interactive visualization and intelligent exploration of time-oriented clinical data," *Proceedings of Intelligent Data Analysis in Medicine and Pharmacology, Protaras, Cyprus*, 2003.

[14] T. Gschwandtner, W. Aigner, K. Kaiser, S. Miksch, and A. Seyfang, "Carecruiser: exploring and visualizing plans, events, and effects interactively," *2011 IEEE Pacific Visualization Symposium*, 2011.

[15] M. Pohl, M. Rester, and S. Wiltner, "Usability and transferability of a visualization methodology for medical data," *Symposium of the Austrian HCI and Usability Engineering Group*, 2007.

[16] W. Horn, C. Popow, and L. Unterasinger, "Metaphor graphics to visualize icu data over time," *Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP-98)*, 1998.

[17] S. Ohmann, C. Popow, and M. Lanzenberger, "Link vis: A novel tool for visualizing and analyzing psychotherapeutic processes." 2006.

[18] T. Munzner, *Visualization Analysis and Design*. CRC Press, 2014.

[19] A. Rind, T. D. Wang, W. Aigner, S. Miksch, K. Wongsuphasawat, C. Plaisant, and B. Shneiderman, "Interactive information visualization to explore and query electronic health records," *Foundations and Trends in Human-Computer Interaction*, vol. 5, no. 3, pp. 207–298, 2011.

[20] F. M. C. N. M. Nunes, "Multiviz: Data visualization of patient records," 2017.

[21] K. D. Eason, "User-centred design: for users or by users?" *Ergonomics*, vol. 38, no. 8, pp. 1667–1673, 1995.

[22] C. B. Stetler, M. W. Legro, C. M. Wallace, C. Bowman, M. Guihan, H. Hagedorn, B. Kimmel, N. D. Sharp, and J. L. Smith, "The role of formative evaluation in implementation research and the queri experience," *Journal of General Internal Medicine*, vol. 21, no. S2, 2006.