Collaborative Project Planning Using Large Format Displays

BY

TOMMASO MASSARI
B.S., Politecnico di Milano, Milan, Italy, 2016

THESIS

Submitted as partial fulfillment of the requirements
for the degree of Master of Science in Computer Engineering
in the Graduate College of the
University of Illinois at Chicago, 2019

Chicago, Illinois

Defense Committee:

Andrew E. Johnson, Chair and Advisor

G. Elisabeta Marai

Franca Garzotto, Politecnico di Milano

ACKNOWLEDGMENTS

Anzitutto thanks to me. Without me this thesis would not have been possible, and thanks to my support, I would have been lost without me. Ma also thanks al Mami e thanks al Papi. Poi thanks anche a alla sista e al little bro. But anche a tutti i bros di Chicago e a tutto il macello che abbiamo fatto.

TM

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

D3	Data-Driven Documents
DOM	Document Object Model
LCD	Liquid-crystal display
LVD	Large Vertical Display
OLED	Organic Light Emitting Diode
SAGE2	Scalable Amplified Group Environment
SVG	Scalable Vector Graphics
UIC	University of Illinois at Chicago

SUMMARY

Large format displays' popularity is constantly increasing in universities and workplaces due to their prices showing a constant decrease over time. Many studies have demonstrated their benefits on several kind of tasks, from navigation of 3D environments to complicated tasks that require high cognitive load, and these studies have demonstrated how they can increase productivity and the quality of work in multi-user collaborative environments.

I am studying the general problem of using large format displays in a multi-user collaborative environments with the specific problem of university study course planning. The traditional approach is represented by a regular whiteboard that will be compared with a project planning/scheduling software developed for large format displays.

The purpose of this study is not only to confirm previous studies' results that claim that the usage of large format displays in a multi-user collaborative environment can increase productivity and work quality, but also to evaluate and understand how their usage influences users' cooperation and collaboration while performing a complex collaborative task. We will study how the subjects actually interact, and how different is their interaction, both with tools and other subjects, compared to the traditional approach.

I developed an application for project planning that runs on SAGE2, an open-source middleware offering the possibility to exploit a large format display of any resolution and size. It also allows multiple users to control it at the same time, connecting to it with a regular web browser interface and a laptop, working in a shared environment.

SUMMARY (continued)

I will show four simple techniques to implement a general project planning application for large displays and I will show four different possible realizations using the same techniques, with a detailed overview on the one created for the user study.

An innovative approach to collect data from the user study will be shown and explained in detail. Such approach improves and automatize the qualitative data analysis that must be conducted in similar studies. It includes space tracking of participants and automatic coding.

Results show that the developed application is not missing any critical feature and it could be a good substitute of the traditional approach. Data suggests that collaboration behaviors could be influenced by such technology, and some slight differences on collaboration were found, but not enough to constitute statistical evidence. In a similar way, no statistical evidence was found about improvements in performance over classic whiteboard technology.

CHAPTER 1

INTRODUCTION

In this chapter I will introduce the reader to the topics touched in this thesis, providing a brief description. I will also state the objectives, and finally provide an overview on the structure.

From the beginning of the technological revolution and the internet age, collaboration was bound to change forever. Starting with electronic mail, instant messaging, audio and video calls, communication has profoundly changed, in every aspect: the way people communicate, how often they do it, and where they do it.

In a similar way technology has been changing the way people collaborate, in many different forms and environments. Sharing documents of every kind has never been easier, and now real time collaboration on the same document, co-located or remote, is possible. One of the technologies that has emerged recently, and that is a promising way to change and improve a more complex kind of collaboration, such as in conference rooms or larger environments, are large format displays.

Such technologies have been partially studied as a possible enhancement to collaboration, but it's still unexplored for the most part. This is due to the high prices and low specific software development. But the last years has shown a constant drop in prices of displays, including larger ones, making them more popular.

1.1 Thesis Topics

My thesis topics include large format displays, multi-user environments, collaborative tasks, project planning, and collaboration behaviors. The same topics are explained with detail in the Background section.

Large Format Displays

Large displays are displays of larger dimensions than standard ones, but it's the context that defines whether a display is a large one. The word "large" doesn't necessarily refer to resolution but to physical dimensions. Their constant price decrease over time has made their popularity grow, especially in workspaces and schools.

Resolution, though, is an important parameter of large displays, indeed physical dimensions and resolution generally grow in tendem. Displays treated in this research and large and high resolution.

Multi-user collaborative environment

The environment being studied is composed of multiple people collaborating to reach a common goal. Collaboration has been proved to be enhanced by large displays in many areas.

Project planning

Project planning is a part of project management, it's a complicated problem that involves the scheduling of tasks and the allocation of resources in the future.

Collaboration Behaviors

In this research I'm interested in understanding the different behaviors that people show while collaborating. Specifically I want to compare their behaviors while working with and without the help of technology.

1.2 Thesis Objective

This thesis objective is oriented towards the human-computer interaction discipline, and it has two objectives.

First, the research will contribute to the literature that studies large format displays benefits in multi-user collaborative environments, enriching it with a study on the specific field of project planning and scheduling. Furthermore, the study has been conducted with a new approach oriented towards understanding how such technologies affect collaboration and human behavior.

The purpose of this study is, not only to confirm previous studies' results that claim that the usage of large format displays in a multi-user collaborative environment can increase productivity and work quality, but also to evaluate and understand how their usage influences people's cooperation and collaboration while performing a complex collaborative task. We will study how subjects actually interact and how this interaction differs, both with tools and other subjects, compared to the traditional approach.

Secondly, I will provide a set of simple techniques to implement a general layout of a project planning application for large and high resolution displays.

1.3 Thesis Structure

Chapter 2 provides some background knowledge and useful information to better understand the topics treated in this thesis. It introduces the reader to the advantages that large displays bring, as well as the disadvantages, to the scenario I'm dealing with and to project planning.

Chapter 3 presents some meaningful work related to this study. First, studies on large displays, summarizing what are the lessons learned from the previous research, and finally some studies on collaboration, and on a similar data analysis style that is performed in this research.

Chapter 4 first provides a set of techniques that allow one to build the structure of a general project planning application. Secondly, it provides the description of the implementation of the one created for the user study, showing all the functions and features. Finally, it shows some examples of other implementations using the same techniques.

Chapter 5 provides the description of how the user study was conducted as well as results. First, a description of the setting, materials and the task assigned to participants. Finally, it shopws the results of the analyzed data.

Chapter 6 contains a summary of the thesis, findings and conclusion.

CHAPTER 2

BACKGROUND

In this chapter some background knowledge is provided on the thesis topics introduced in section 1.1

2.1 Large Displays

There is no precise definition of what large displays are, with this expression it is intended a display of any technology, such as LCD or OLED or projection based, of large physical dimensions. So, we have to take into account the context where the display is located to define whether the display is a large one. A 27 or 32 inch display can be considered a large display in a regular office for an employee, but not enough to be considered such in a context such as a classroom or a conference room. In this case we would need a 100 or 200 inch display. The kind of large display we are dealing with are oriented to scenarios similar to the second example, thus displays of dimensions of over 100 inches.

Time has an influence on the definition of large display, indeed over time display dimensions has shown an increasing trend. For instance, the median size of TVs in the early 2000's was 34 inches and became 46 inches from 2004 [1] and increased further in the recent years to 55 and 65 inches as a standard.

---

[1]https://flowingdata.com/2009/09/23/tv-size-over-the-past-8-years/

Resolution, instead, does not take part in the definition of large displays. But it's worth mentioning that, as display size, it has increased over time. In fact, in the early 2000s the standard for TVs and monitors was SD (standard definition, 480p) or, the DVD quality, to the Full HD (1080p) or, the Blu-Ray quality, and currently turning to 4K (2160p). A higher resolution does not translate in a larger screen, but either in increased working space or increased sharpness, or a combination of the two.

Prices of displays have been decreasing over the years, but large displays are expensive. Thus, often, a combination of multiple smaller displays are combined to create a large one. In this case bezel width becomes an important parameter to take into account. Generally it's preferred to keep the value as low as possible, even if the artifact doesn't affect user's performance and attention as demonstrated by studies [1] [2]. In fact, there are other studies that suggest that bezels can be even useful, as they can be used as a guide to position windows inside the grid that they form [3].

2.1.1 Advantages of Large Displays

Large displays have great advantages over standard ones. First, the management of multiple applications is easier thanks to the larger space, and it's possible to display at the same time multiple windows and thus, multiple applications. The user avoids loss of time changing windows and this leads to improved productivity. Many studies have proven that even just a dual monitor working position leads to improved productivity [1] [4]

---

[1] https://www.dell.com/downloads/global/products/monitors/en/dual_monitors_boost_productivity_whitepaper.pdf

Studies show that large displays increase performance of users performing navigation tasks in 3D environments. Indeed, thanks to the wider fields of view, peripheral vision has been exploited by subjects during the experiments. Search and navigation results ar improved because users were able to see more of the environment at any given time [5] [6].

Other studies [7] show that large displays contribute in user's sense-making. While the understanding and making conclusions is a human task, technology can help and provide support on searching, filtering and visualizing information. Large, high resolution displays are shown to be a better solution than regular sized displays in such tasks, giving the users more space and thus more awareness on information, helping creating connections and relationships between documents.

It has been demonstrated [8] that thanks to the larger amount of content displayed, large format displays are a solution that help users solving complex tasks, when a high cognitive load is required. The key difference is the capability of having at the same time, on the same view, a global overview and details. This study is particularly meaningful to this thesis since a complex task, such as project planning, is a main topic of the research.

2.1.2 Drawbacks and Challenges

Large displays are an interesting and relatively new technology but it doesn't bring only benefits. Indeed, some problems arise along with the increase of the display size, and a new field opens with new challenges.

Some of the most common problems that users encounter are losing track of their cursor, and window management problems, for instance, moving, resizing, and placing multiple windows in the preferred position becomes a time consuming task itself. Moreover, physical distance problems occurs, indeed, distances from users to the furthest windows or the furthest icons are too large and it becomes hard reaching them or pointing at them with precision. [9]

Software itself is an issue as well. Generally developers when developing applications and graphic interfaces take into account regular display sizes, and generally it happens that the same interface, when run on a large display, doesn't scale properly and some elements of the interface, such as icons or notifications, appear in non-intuitive positions or even randomly. Thus, often developing an application for large displays requires a different approach that leads to different design choices and solutions. The same way smartphones, and in general small displayed devices, require a different approach and a different interface, large displays do as well.

A perfect example are websites: they have a specific layout for regular sized displays, for instance, when browsing from a laptop or from a common desktop computer. With the introduction of smartphones and their large diffusion due to their popularity, developers had to reinvent the layout and create a specific one for such small screens. Nowadays the majority of websites have their smartphone version counterpart. In the same way large displays need a new third layout of the same web-page, and application interface.

With new technologies always new challenges arises, and the world of large displays is still not completely explored. There are paradigms to follow to build a smartphone interface but there is none to build a large display one. [10]

Looking into the near future, the next technological challenges [11] in the field are :

- Seamless tiled displays: shrinking to non-visible bezels when dealing with tiled displays. Nowadays it's possible to find 65" displays with less than 2mm wide bezels, and direct view LED tiles have virtually no bezels.
- Easily reconfigurable large high-resolution displays: even if it's more simple than before, building and modifying a tile display is still a complex task. It should become as easy as a "plug-and-play" type of device.
- A New and effective interaction techniques: even if traditional input devices do work with large displays, they are often not optimal and less intuitive than usual. The scientific community is still researching on this field.

Moreover, to overcome the most common problems with multiple windows, a new window management systems specifically designed for LVDs are needed.

2.2 Multi-user Collaboration

The second topic of this thesis is collaboration. I will deal with a multi-user environment, i.e. an environment where more than a user at the same time is operating. In this section I will give a background on collaborative software and multi-user environments.

There are three main types of multi-user environments: collaborative, competitive, and independent. The context in which this thesis takes place is collaborative, i.e. an environment in which users have the same task and they share the goal, and are working in collaboration to achieve it.

Figure 1 shows all the possible scenarios. This thesis' scenario is multi-user collaborative environment with mouse and keyboard input, and co-located location.

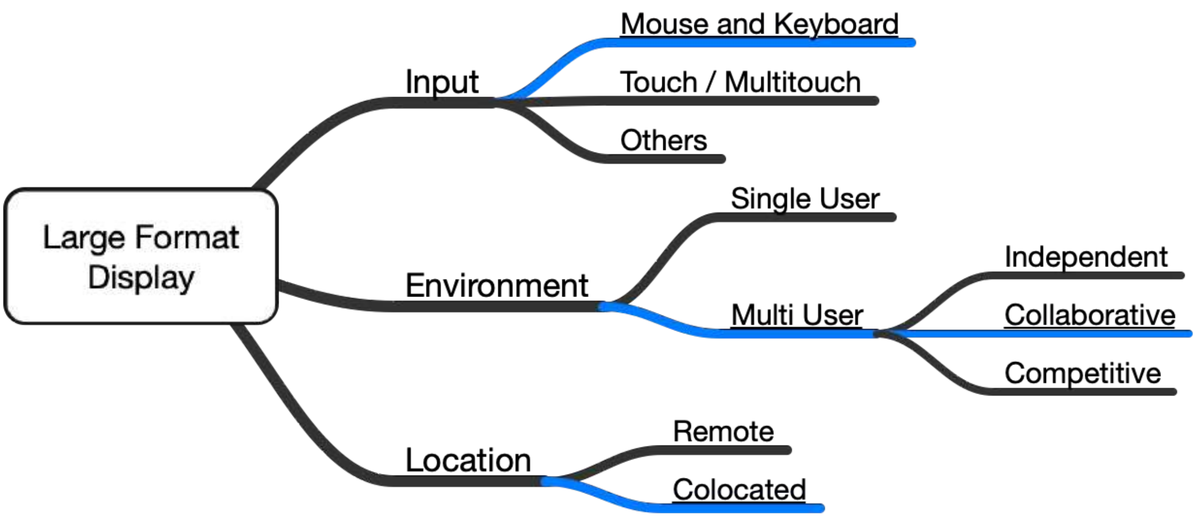

Figure 1: Possible scenarios of large format displays. Highlighted in blue the specific one of this thesis.

Many studies have been conducted on this topic, and I cannot fully take into account all the studies in this field because the context i'm dealing with is more restricted. Indeed, there is a difference between the studies conducted on horizontal displays and vertical ones, the latter being our interest. The difference is less subtle than it appears, indeed, movements and the position of the cooperating people are important parameters. In a horizontal environment people are generally sit and tend to stay in fixed positions, while on a vertical display movements are more common, and thus, this leads to different dynamics of collaborations. However, results of such studies can still be taken into account [12] [13].

Indeed, as proved and explained in a study conducted by R. Ball and C. North [14] users tend to physically move in front of LVDs, and they generally prefer to move their bodies instead of zooming and panning. It turns out that when dealing with high-resolution vertical displays there is naturally more physical navigation performed by users, then in their low-resolution counterparts, and more virtual navigation in these latter ones.

Collaborative application are generally more complex than single user ones, and they need a different approach and development solutions. [15]

2.3 SAGE2

SAGE2 [16] is an open source software that offers a solution to the described problems of large format displays. It's a middleware that enables the management of displays with non-standard proportions and resolutions. It's the system that I will employ in this research for building the project planning application that I will cover with details in section 4

SAGE2 allows the combination of multiple displays into a single space that can be used by the user as one single screen. Moreover, it enables a multi-user environment that can be both co-located and remote, with built-in applications though which users can share any kind of document, images and videos, and even entire applications.

SAGE2 has a client-server structure where the server has its own address and to access it, any web browser can be used just by connecting to the server's URL. There are two different types of clients: input clients and display clients. Display clients are generally connected to the large display and they show the actual shared screen. Input clients are generally accessed from laptops and are used to control applications.

SAGE2 includes many application to perform basic tasks, such as reading PDFs, viewing images and videos, maps and clocks. Furthermore, it allows the creation of custom application using javascript, with some restrictions on inputs, since it has its own input system. Inputs come from input clients, then they are sent to the server that is responsible of the communication to the corresponding application on all the display clients.

2.4 Project Planning

In this section I will introduce the project planning problem, the problem object of this thesis user study.

Project planning is a part of project management, during this phase proper documentation is written to support successful project development. Planning requires several inputs, that vary according to the project, such as "conceptual proposals, schedules, resource requirements and limitations and success metrics"[1]. Project planning begins with the project itself and planning accompanies the project throughout its development, sometimes to the end. Indeed, often it's necessary to review the plan, reschedule tasks, or reallocate resources especially in large and long term projects, as it could be the creation of a new machinery. Generally, Gantt charts are the form where all this information is displayed and integrated, or other analogous types of scheduling charts, to provide a project overview.

Project planning is widely demonstrated to be a relevant phase for large projects success. In the literature several studies have been conducted and an important correlations between planning and success has been found. [17] It has been shown that an effort spent on planning of 20% to 33% is highly correlated with a subsequent project success.

---

[1]https://www.techopedia.com/definition/14005/project-planning

Software tools for project management exist and in the literature it's even possible to find a comparison chart in order to select the one that fits the most user's needs [18]. All these tools have been developed for regular desktops and not for large format displays. Project planning is just a part of project management and it requires people to process a large amount of information. Two factors are to be taken into consideration: first, large displays popularity is growing in offices and conference rooms, and second, one of the main advantages of LVDs is the higher amount of information that can be displayed at once. For these reasons a large display should be a better fit for this specific task.

No research has been conducted on this specific topic and no software has been specifically developed for such task for large displays. Today project planning is often performed manually, through the use of large whiteboards, or wall-long paper sheets and sticky notes, making this kind of approach still the most common, even in big companies. Such an approach incurs all the disadvantages and discomforts that a manual approach brings: difficulties in making changes, long writing and transcription sessions, lack of clarity of information displayed and frustration.

In my user study I will use a specific instance of the project planning problem that is a planning a new course of study for a bachelors degree.

My objective is not to create an automated software to provide scheduling and planning solutions or helping the human counterpart in finding a better solution through the automation of some parts of the procedure. My goal is to study human behavior, understanding if and how technology, in this specific case a large format display, influences or affects collaboration and understanding, and whether such technology could be an advantage over the traditional approach.

To have a better understanding of the problem we can take into consideration other studies on the theoretical analysis of the time scheduling problem. An important contribute in this field was given by Garey and Johnson in their book [19]. The problem analyzed is the time table design problem, a slightly different instance of the planning problem, but they belong to the same type of problem, making both, and all the related problems, of the same computational complexity. In the book it's shown to be NP-complete[1] making it a non-trivial problem, both for humans and machines. This is the complexity to expect from a typical problem of such kind.

2.4.1 Why the Traditional Way is Still the Most Common Approach

As mentioned in this section, the project planning problem is still, most commonly, solved with a traditional approach, using whiteboards and sticky notes instead of technology. The reasons to justify this facts are mainly two.

Requirements and user needs

This kind of problem can highly vary depending on the user and their needs and requirements could be completely different from one to another. There are no existent standards and it's too complex to create them in order to make a standardized version of the problem. For this reason many companies are not able to find a software that fits exactly their needs. Each company

[1]NP-complete problems are all those problems that cannot be solved by an algorithm in a polynomial time.

would need an ad-hoc solution to fit their needs. For example, some users may find some time slots or some resources more important than others, or they might need to display different kinds of information, or when too onerous, to decide what information is relevant, or even add more "dimensions" to the layout, such as color coding.

Large information amount

Often a session of project planning requires a large number of people to be involved, and a large amount of information to be processed. The same information also has to be visualized, and people need to have a global view of the problem. Regular software, that is designed for standard screen sizes and resolutions, is not able to display this large amount of information at once, failing to provide a global view.

2.5 Qualitative Data Analysis

Chapter 5 shows the user study I performed. But before facing this topic, I need to give some background on the kind of analysis I performed. The user study consists in making groups of participants solve a complex task. While the subjects are working, I collected data from different sources, such as audio and video recordings and their head orientation and position.

Data sources are of two types: qualitative and quantitative. The latter is data that can easily compared, indeed, generally, it consists of numerical data, but I will deal mostly with qualitative sources. Qualitative data, also known as descriptive data, consists of non-numerical data that captures concepts and opinions. For instance, our audio and video recordings are part of this typology of data. When dealing with qualitative data, a different approach of analysis must be performed, and it is known as qualitative data analysis.

Qualitative data analysis is the process of examining this kind of non-numerical data, our qualitative data. It reveals patterns and themes in the data sources. Thanks to this type of analysis I am able to interpret the data, organize it, find patterns, and interpret it according to my objective research. For its underlying nature such kinds of analysis, though based on some ground rules, does not follow a rigid process. In fact, this approach is highly dependent on the researcher and the context of the study.

Data Coding

When performing qualitative data analysis, a common approach is to code data. A code is a concept, a property, or a pattern that is meaningful to the research involved and that you can extrapolate from the data sources. Coding the data is categorizing it, and transforming it into easily understandable concepts for a more efficient data analysis process. Codes can be derived from theories, from relevant research findings or from research objectives.

Section 5.8 will explain what codes have been chosen for this research and comment them in detail.

CHAPTER 3

RELATED WORKS

In this chapter I will discuss and analyze the most relevant studies related to the topics of this thesis.

3.1 Studies on Large Displays

Even if the field of large displays is relatively new and still, for the most part, unexplored there are several studies on large displays.

3.1.1 Dual Display and Large Display

Studies start from more simple, but more common, configurations, such as a double monitor workstation. It is shown to improve productivity [4], leading to slightly faster results and reduced error rate [20]. Also, the latter study compares a traditional screen with a larger wide-screen, showing the latter gives better performance and usability. The study concludes that users experienced measurable gains in productivity and the work finally was judged easier to perform, both with the wider display or the dual monitor one with respect to the standard one. This study does not test the same large displays I'm dealing with in this thesis, but the results point to the same direction, in fact they suggest that increasing the display space user performance and productivity increases too.

Consequently, there are studies that try to fill this gap and compare directly standard displays with large ones. M. Czerwinski [21] conducted a study through which it is demonstrated that user satisfaction and productivity were significantly higher with large displays instead of a traditional one. This study, though, is confined to single user tests, and did not test collaborative work nor was it heavily oriented toward multitasking tasks. In the study the lack of large display designed software is mentioned, through which it would be possible to achieve an even higher progress in productivity and user satisfaction.

Another study conducted by Bi X. and Balakrishnan R. [22] also compares LVDs with a dual monitor in a daily work environment, resulting in users prefering LVDs. They are shown to benefit specific tasks that require multi-window or rich-information. In the study it's shown to enhance users' awareness of peripheral applications, and to offer a better overall working experience. The study leads to clear results: users' satisfaction generally is increased by switching to a LVD and productivity increases as well [23]

Productivity is also proven to be improved for basic visualization tasks in finely detailed data [14]. Using physical navigation over virtual navigation helps in keeping a global view of the context and it helps in comparing data faster.

3.1.2 Other Studies on Large Displays

In this section some interesting studies in single user environments are discussed, and in section 3.4 I will discuss studies in multi user environments.

Large format displays have been shown to enrich productivity or user performance in many different tasks and scenarios. As discussed in section 2.1 LVDs are shown to provide advantages in navigation in 3D environments [5] [6], sense-making [7] and solving complex tasks that require high cognitive loads [8].

Other studies [24] try to understand what are the differences in learning and cognition while using a large display, and how differently information is perceived and processed depending on viewing distances and viewing angles, and on their position, whether static or dynamic (as users are able to freely move). The results can be taken into account when designing an interface for large displays. Results show that the accuracy was the same but time was decreased for the static position, and that area, distance and length impact accuracy, though length is less affecting.

Another branch of large format display research concerns large displays with multi touch technology. This interaction type is inherently different from the one inherent to my study, but it's interesting that studies go in the same direction, confirming even further the effectiveness of large displays in learning and cognitive functions such as attention and memorization [25]. A user study was conducted specifically in learning and cognition for digital natives with positive results on the effectiveness of such technology [26] [27].

3.1.3 Lessons From the Research

Moreland tries to evaluate the most important lessons learned from research on large displays [10]. The first lesson is about building tiled displays: technology and techniqueas in building a large display from multiple smaller displays has now matured. With the decreasing price of the hardware it makes it more affordable and feasible to sell these displaus as a single embedded solution. A second lesson is about visual information limits: higher information content, such as more resolution, does not always translate in an increased amount of information acquired by users. In fact, large displays enlarge even more the gap between information displayed and information acquired. Unfortunately, on how to properly use a large display, there is still research to be done, to discover which design principles, if any, to adopt, and for which application, and which are the "sweet spots" between resolution and size. A third lessons is about input: there is still no specific human-interaction interface for LVDs, and standard ones, such as keyboard and mouse, or touch screen, are subject to inefficiency and they can impede interaction. A new human-computer interaction mode needs to be invented and to become a standard in the future. Currently, the choice of input device must take into account the kind of application the user will deal with. According to the display size, context, number of users, distance from the screen, a different input interface could be chosen over another one, and for every application the best choice could be different.

3.2 Collaborative Tools

In this section I will give an overview on some of the most common collaborative tools available. Collaboration, co-located and especially remote, has become more popular lately, a reason is the growth of the "cloud". Cloud technologies allow the storage of the information directly on the servers of the service provider. In this case, user always access the latest version, since the files are always updated. This leads to a greater ease of use respect to approaches that have different copies of the files on all collaborators local machines.

3.2.1 Collaboration on Office Documents

The most common application for office documents are provided by Google, Microsoft, and Apple. The first are generally light editors providing less features then the last two.

Text Documents

Google Docs is the on-line tool provided by Google for text documents. It allows users to connect at the same time time and work on the same file, and it provides real-time view of the changes made by other users, including a visual feedback on where are the remote collaborator's cursors. In the version for businesses it is also possible to access the documents while off-line [1].

Microsoft Word is a tool that has to be installed on the user's machine but it is richer in features and allows to work on the files while off-line. Documents, until the 2016 version, had to be synchronized by users manually, but from the 2016 version on a similar real-time experience to that Google Docs has been introduced. [2]

Apple Pages is the counterpart of Microsoft Word for Apple ecosystem and offers similar functionalities. It provides as well an analogous real-time experience to Google Docs [3].

Spreadsheets

Similarly to text documents, Google's tool is on-line based and doesn't need a software to be installed on the user's machines, while Apple and Microsoft tools do, but they provide a richer features experience, and can be used while off-line.

---

[1] https://gsuite.google.com/features/?utm_source=docsforwork&utm_medium=et&utm_campaign =refresh&utm_content=header&hl=en

[2] https://www.laptopmag.com/articles/co-edit-document-microsoft-word

[3] https://support.apple.com/kb/PH26109?locale=en_US&viewlocale=en_US

Google Sheets is the Google tool for real-time collaboration on spreadsheets, and Microsoft Excel and Apple Numbers are the richer alternatives. Analogously to text documents Excel provides real-time collaboration from 2016 version on, as so does Apple Numbers.

Presentation

Analogously to text documents, and spreadsheets, Google Slides is the on-line tool and provides a similar real-time collaboration as presented above, and Microsoft PowerPoint and Apple Keynote are application based and feature rich, but allow a similar real-time collaboration from recent versions.

Google Forms

This is tool that regular office suites doesn't provide, and allows to get answers, make surveys, or Q&A easily and remotely. Real-time collaboration is provided for forms writing.

3.2.2 Sketch and Whiteboard Collaboration

ConceptBoard is an on-line tool that provides the possibility of real time collaboration on a whiteboard [1]. It visually provides instant feedback of other users changes and it provides the view of collaborators pointers, in a similar way that SAGE2 does. It allows to share almost any kind of document, form text to images and videos, but it doesn't allow the users to sketch, as on a real whiteboard.

---

[1] https://conceptboard.com/features/real-time-collaboration/

Binfire software [1] is another solution for collaboration and project management. They provide a virtual whiteboard that allows real-time collaboration, again without the possibility of sketching. They provide a suites of software to support project management too. The "task management", "Gantt Chart", and "resource loading" parts are the ones my application is related to. This software though has not been designed for large displays.

Twiddla [2] is another example of tool that allows real-time collaboration and sketching on a virtual whiteboard.

Many other similar tools exist, one worth mentioning is Realtime Board, interesting for its sticky-notes feature and quick diagrams among all.

3.2.3 Collaboration on Project Management

There are many solutions to project management and some of them allow collaboration. Generally though these are suites of software that are thought to support all the phases and all the aspects of a project. My focus is only on the one regarding planning and scheduling tasks and resources. Furthermore, none of these have been developed for large displays.

Tools worth mentioning are: ProjectPlace[3] and Wrike [4]. They both offer Gantt chart views and collaboration, teamwork organization and To-Do lists, reports and workflow management.

---

[1]https://www.binfire.com/project-collaboration-software/

[2]https://www.twiddla.com

[3]https://www.projectplace.com

[4]https://www.wrike.com

These are just two examples, but the possibilities and choices are many, depending on company size and category.

Some state-of-the-art tools for project management are: Microsoft Project and Oracle Primavera. Those are complete piees of software that provide all the tool necessary to follow the lifespan of a project, from document archiving, to time tracking and resource allocation.

Microsoft Project provides real-time collaboration in an analogous way of Microsoft Word, Excel, and PowerPoint do, allowing users to work remotely on the same files and viewing other users editing and even their cursor position in real-time.

3.3 Studies on Collaboration

In this section I will report some interesting studies on collaboration, not specifically related to large vertical displays.

The first study is about co-located collaboration, with the difference on the display orientation, as this study is performed on a tabletop display [28]. They tested the effectiveness of a tool for collaborative work with a large number of text documents, named Cambiera, and found that co-located collaboration working face to face is an effective way for solving complex tasks. The interesting part of this research is their definition and categorization of eight different kind of collaboration types between team members while working on a complex task. I will discuss their coding decisions and how they relate to my user study with more detail in Section 3.5.

Another interesting study compares large vertical displays to large horizontal displays [29]. The results of the studies are that to solve a complicated problem solving task both large screens helped collaboration and solution finding against a regular computer. Interestingly, the horizontal display was found to help more than the vertical one on solution finding and collaboration. Indeed, using the horizontal display users changed roles more frequently, explored more ideas and more easily followed other participant's work. They evaluated that the users position around a table made social interaction and information exchange more comfortable. But when asking directly, some of the subjects some were of the opposite opinion, as they found interaction in the horizontal table confusing and chaotic. The reason is the more dynamic democratic setting of the horizontal table, where anybody at anytime can take the "leader" role, compared to a setting where the "leader" is more defined and with less frequent changes.

The interesting part of this study is that how people cooperate has been studied, specifically in role change, division of labor, and ideas exploration, in the same direction to the user study I will conduct. They mention the "PRACTICE" framework, a set of questions and issues they used to interpret the collected data.

In a second study of the same research, they examined how both configurations together, an environment where both a vertical and an horizontal display are present, affects collaboration. This resulted in a greater division of work and fixing of roles.

3.4 Collaboration on Large Displays

In this thesis we are interested in understanding behaviors in multi-user collaborative tasks in a co-located environment. Several studies have been conducted on collaboration while using large displays, and in this section I will report the most meaningful ones.

Interesting behaviors appeared in the study conducted by [30]. This study involved groups of people collaborating to elaborate a large amount of data in text documents form. The purpose of this study was to investigate users' spatial strategies in the management of the virtual space formed by the large display. Interestingly, users used the screen as a means to aid memory, organizing information into clusters of text files. The space were used to make connections and recall information. The study also compares two different kinds of software and their behaviors in large displays: data-centric and function-centric tools. The first allowed users to better utilize virtual space and thus, allowed a better extrapolation of information.

Another study [31] was conducted on collaboration while using a large display with multi-touch input type on a complex task such as business process modeling. The study interestingly remarks how the collaboration in such a task was modified by the technology, compared to a regular session of process modeling. Specifically, people were dynamically creating subgroups on demand, behavior that rarely happens in a regular session. Another advantage that appeared was related to problem comprehension: users were able to track changes, even if performed by other users, more easily. This study is worth mentioning since understanding differences in behaviors of the same collaboration task is an objective if this thesis.

A further contribution to collaborative studies on large displays is [32]. First, this study contrasts previous studies that were showing difficulties in collaboration on a wall display, stating that large format displays with multi-touch input can support group work and even different collaboration styles. To provide such results, the user study was conducted assigning users a complex task of problem solving over a set of documents. Users performed sharing of the spaces, physical navigation to different display areas, and switch fluidly between parallel and joint work.

Another interesting work on collaboration on large displays has been conducted by Filippo Pellolio [33]. In his research a software version of a whiteboard has been developed for large displays and a user study on collaboration has been performed. The whiteboard allows the users to sketch as in a real board through the use of multi touch. In this case the input style differs from my research, and also the location, being remote instead of co-located, but the results can still be taken into account due to the many similarities of the approaches. The user study showed no significant performance improvement of the display counterpart, but it helped collaboration by creating a different and more collaborative environment.

3.4.1 A study on scheduling

Davide Tantillo has conducted a study on the field of large format displays applied to scheduling [34]. A Conference Scheduler application has been developed and in his work he conducted a study involving groups of 4 people. His work is particularly related to mine considering the many similarities in the nature of his work. He conducted a study that analyzes human behavior on large displays and was trying to identify similarities and differences between the new and the traditional approach, as well as trying to find improvement in performance and quality of work.

The main difference is in the type of input, touch screen in the case of Tantillo's work versus mouse and keyboard in this work. This study has important similarities to mine and the assumptions and results can be compared.

Tantillo conducted a user study on the StickySchedule [35] application. The proposed application was a tool for the generation of a schedule for a scientific conference. It allowed multi-user interaction and it was developed for large displays. The purpose of the application was to organize large conference scheduling. In their study the technological approach was represented by the large display running the latter application and the traditional approach was represented by a whiteboard, in a way that is similar to the user study I will conduct.

In their conclusion they state that the application does "provide a similar look and feel to the traditional whiteboard approach for conference scheduling" [35]. These results suggest that such an application could be a good substitute to the traditional approach.

The study, though, didn't show any performance improvement of the application against the traditional approach. The quality of results remain similar and the time spent was comparable as well. Also, no evidence of less frustration signs was found in the technological approach.

3.5 Studies on Qualitative Data Analysis

In this section I will present some interesting studies on collaboration that performed the same kind of analysis that I will employ in the user study of Chapter 5. As explained in the Background, I will need to find codes to categorize actions of the participants, to better understand their behaviors during collaboration. In order to find the most significant codes that allow me to capture the most relevant information about collaboration, I can look at the literature.

The first study was conducted by Jakobsen and Hornbæk [32], the same study cited in Section 3.4. In this study, similar to this thesis case, audio and video has been recorded and the position of participants has been tracked. Furthermore, the tested subjects were couples, making this case even more similar to this thesis'. Interestingly, to describe the subjects collaboration they decided to code separately verbal communication and visual attention. Furthermore, they made a distinction between physically based codes and socially based codes, the first easy to identify while the second rely on abstraction and interpretation. Regarding the visual attention, an interesting coding approach is that they provided different codes for the area of the screen participants were looking at, for instance if they were looking at the same area of the screen or different areas, or if they were looking away or at each other. The combination of 3 different types of codes, visual attention, verbal communication, and proximity give insights on the level of collaboration, whether tightly coupled or loosely coupled. Another interesting aspect of this study is the involvement of proxemics. Proxemics is the study of how people interpret the space around in relation to other people, dividing the space in zones, depending on the distance to the subject. Zones go from intimate, to personal, to social. They included in their study the proxemics element, trying to find patterns on distances between subjects and their collaboration.

Another study conducted by Isenberg et al. [28] shows a relevant example of coding whose results can be used by my work. As before the setting is on a tabletop display, but this doesn't make their coding less relevant to my study. They decided to have a unique set of codes instead of separate sets as the Jakobsen and Hornbæk [32], previously discussed. The interesting part of such schema is that the eight codes are ordered, from the one that conveys the most "loose collaboration" to the one that conveys the most "close collaboration". Their codes are respectively in the same order: "disengaged", "working on different problems", "same problem different start", "same problem different documents", "same information different views", "same view", "view engaged", "discussion". As we can notice, this unique set of codes combines together visual attention and verbal communication.

A third study I can take into account has been conducted by Tang et al. [36]. Once again, the setting involves a tabletop display, investigating collaborative couples. Similar to the Isenberg et al. [28] study, they decided to use a single set of codes. They identify six codes, combining together visual attention and verbal communication. They divided the groups in two parts, associating to 3 codes the meaning of "working together".

Finally, the previously cited work of Tantillo [34] is a very interesting source for codes and coding style due to the many similarities of my user study to his. He was interested in analyzing human behaviors in the same way I am. He's aim was to find differences and similarities in terms of human behaviors. To do so he analyzed the current literature to find similar studies to get inspiration for his data coding. His conclusions present significant similarities to mine, thus, his coding decisions can be taken into account while creating appropriate codes for my case. The most interesting codes for my research include visual attention, distances between subjects and tools, and group shape, all with the proper modifications to fit my case.

We've seen in this section four different approaches to coding that will be of inspiration for my user study.

CHAPTER 4

IMPLEMENTATION

In this chapter I will introduce the readers to the project planning application that has been developed to conduct the user study described in chapter 5.

Such an application is the result of an idea that came from a related work activity in the field of project planning. Caterpillar needed to substitute their current traditional approach with a software one to overcome all the difficulties of dealing with large planning manually.

The application is entirely written in Javascript with the help of some external libraries. It's designed for large vertical displays and it exploits their high resolution and large virtual space, but this doesn't limit the utilization exclusively to them, allowing it to run also in regular sizes monitors, keeping in mind that all the advantages that come from the larger space are lost.

First, I will describe some general techniques that combined together can be utilized to create a simple interface for a general scheduling/planning application. Then, I will describe in detail the structure of the one that has been developed for the user study. It is a result of several attempts, and is the combination of all the building blocks that were necessary to build all the features with to respect all the requirements. Finally, at this Chapter's end I present two other implementations based on the same application structures described in section 4.2

4.1 Related Activity

The idea of the course scheduler application used in this thesis for the user study came from a related activity in the field. Caterpillar organizes and manage their project planning meetings entirely manually, and to overcome the common issues of the manual approach and speed up the process they requested a software version, using SAGE2 and large displays.

Caterpillar showed interest in the world of large display when they were introduced to the work conducted on StickySchedule [35]. Such an application combined with large display technology was very promising for switching their project planning approach from manual to a computer version. The StickySchedule application, however, didn't exactly fit their needs and there were differences in the structure behind the problem. Conference scheduling, indeed, can be viewed as a specific instance of project planning, and what they needed was another, different instance of such a category of problems.

The process started with a meeting for the description of the problem and definition of software requirements. I've been introduced to their current approach, with details about the issues of most concern. Their conferences involve about 10-30 people representing the different teams. A wall-long piece of paper is placed on the wall to represent a time-line that can vary from some months to a year. Previously created sticky notes containing all the tasks and milestones are discussed and then placed to form the plan. All the data is then manually transcribed into a spreadsheet. Task number can vary from 75 to over 400.

The points of biggest concern were the long transcription sessions, the readability of sticky notes, the difficulty to perform modification in the structure, such as adding a new lane for a new resource or a team, or to lengthen the time-line.

4.1.1 Requirements

Considering all the similarities that StickySchedule [35] has with such an approach I can also take into account their requirements. Indeed, similar requirements include non-functional ones such as scalability. This could be considered a more general requirement of all large displays applications, indeed the nature of large displays implies many different combinations of resolution, size and aspect ratio. The second similar requirement for StickySchedule is support for local and remote collaboration. In our application I focus on local collaboration, but this could be considered as a requirement for some future development. Further requirements are specific to the constraints of the conference scheduling problem and are not meaningful to my requirements gathering. Some requirements on the virtual sticky notes though can be taken into account: creation of sticky notes, and drag-and-drop functionality.

On the base of the acquired knowledge on the dynamics of real scenario project planning conferences, the gathering of requirements for building the application started. Initially requirements were high level, and a general project planning application was developed. Later, after a meeting and company feedback and impressions on the general version, we reviewed the requirements to better fit their specific needs, and they provided a dataset of sample data taken from a real past conference. With this new information I performed corrections and strengthening of the previous version to better face the real case scenario.

Similarly, the requirements gathering of the study course planning application started first with a more general project planning application, and then strengthened them after expert review. Most of the requirements were then adopted while creating the study course planning application, with little modification due to the slightly different nature of the problem.

In summary, the following high level requirements were extracted after an interview, and successive review with experts. Such requirements are both functional and non-functional:

- CSV import: reading a prepared set of items from a csv file. This allows the session to start from that initial configuration;
- Save: possibility to save the session and export a csv file, making it possible to restart the work from that point;
- Add new: possibility to add a new item, with all the necessary information;
- Remove: possibility to remove old or wrong items;
- Modify: possibility to modify already present items;
- Global visualization: view of all the items in a grid to provide a global view of the progress to users;
- Move: this is a sub-requirement of the Modify one. It adds the possibility to move already present items with drag-and-drop type of interaction. This requirement makes modification of the 2 most important attributes of items (i.e. the two represented into the axes of the grid) faster and in a more natural and intuitive way;

- Pool lane: presence of a pool lane, an area in the graphic interface that contains all those items that don't have a position in the main grid yet, or that users are still working on.
- Time adjusting length: items length in the visualization should reflect and be proportional to their real length in time. For instance, an item that has a length, in terms of time, double of another item should be also visually two times longer than the second one.
- Zoom: possibility of zooming in to have a better view of items. This is useful when there are too many items or they are too small in the global view. Indeed, the item number in real scenarios could grow so much that even a large display would not be enough to handle them.
- Home button: having a way to bring the user to the general global view.
- Collaboration: possibility to collaborate simultaneously in the work in a co-located type of collaboration.

Similar requirements can be used to build the study course planning version of the application, with some modification after the following considerations.

During a session performed with the traditional approach, such as using a whiteboard or a long paper sheet, people need to "initialize" the scenario by performing the following actions:

- Draw the grid for the time-line;
- Write sticky-notes with all courses to schedule, writing down all the necessary information;
- Optionally, assign a meaning to sticky notes colors, for instance, a different color per course category;

Subsequently, users need to perform the following actions:

- create new sticky-notes with new courses.
- correct old sticky-notes if needed, for instance, if one is wrong or incomplete.
- assign courses to a term.
- move courses, assigning them to another term.
- trash sticky-notes
- retrieve all the information necessary, such as days, times, and credit hours from the sticky-notes.

All these actions allow the organization of a course of study plan, and the same set of actions have to be offered into the software version as well.

Since with course planning the number of courses to fit is always limited to 30-50, and a large display (specifically the dimension of the ones provided for the user study) is always capable of dealing with these numbers, the zoom requirement has been dropped because it has been deemed not useful for this specific application.

The study course plan application must provide all the functionalities that are available in the traditional approach. Given all the requirements described previously this constraint is respected.

4.2 Techniques to Implement a Planning Application for Large Displays

Since the lack of a paradigm to follow for building applications for large format displays, as discussed in Chapter 2, in this section I propose some simple techniques to implement a graphical interface for a general project planning application. The techniques are the result of the confrontation and review with experts as explained in 4.1. The same general techniques have been used to develop first the project planning application for Caterpillar and then the study course planning application used in the user study.

Developing an application for large displays implies taking into account some assumptions that are new or different from the usual.

First, the resolution: large displays generally involve higher resolution, due to the larger physical size, and often largely variable, in terms of number of pixels and aspect ratio. Since I'm dealing with such variabilities and high resolutions I needed a solution that allowed me to preserve the image quality. In order to allow the user interface to be independent from size and resolution I employed SVG, a vector graphic commonly used in websites, another example of a resolution independent application.

D3 (version 4) is the SVG library that has been used to draw SVG images and graphs. Indeed, SVGs are composed by an XML file, but, even though it would be possible, SVG images are not created directly modifying such file but through the use of libraries. The choice of D3 among the many existent libraries is due to the fact that it is known for its graph plotting and visualization capabilities.

The techniques showed below are written using JavaScript and they have been developed for a SAGE2 application, but they are completely independent from such environment and they can be used for any application.

4.2.1 Division of the space

Before starting to create the elements, it' necessary to divide the space into areas to dedicate them to the different functions. The grid element is the center of the visualization, and where most attention of the user rests. Thus, it's necessary to dedicate most of the space, and a central collocation to it.

The areas to define space are:

- control area: the area that contains buttons;
- task area: the area that contains the grid and the virtual sticky notes;
- pool area: the area for "work in progress" sticky notes;
- zoom area: the area related to the zooming functions and containing a fixed global view;

For the task area, some margins are created to make space for text and for better spacing. In this case they have a fixed number of pixels but they can be configured with variable measures, depending on screen size.

The first thing to do, is to create a container for the SVG that will occupy all the space of the application, and then divide all the areas:

```
let margin = {top: 30, right: 30, bottom: 50, left: 50};
let svgEx = d3.select("body").append("svg").attr("id","WholeArea");
let windowWidth = element.clientWidth ;
let windowHeight = element.clientHeight ;
let taskWidth = windowWidth * 0.85 - margin.left - margin.right;
let taskHeight = windowHeight * 0.9 - margin.top - margin.bottom;
```

```
let controlAreaWidth = windowWidth * 0.05;
let controlAreaHeight = windowHeight * 0.9 ;
let zoomAreaWidth = windowWidth * 0.9 ;
let zoomAreaHeight = windowHeight * 0.1 ;
let poolWidth = windowWidth * 0.1;
let poolHeight = windowHeight;
```

Listing 4.1: Division of the space

Subsequently it is possible to create SVG groups, one for each area to whom in the future it will be possible to append all the elements belonging to the area:

```
let viewBox = `0,0,${windowWidth},${windowHeight}`;

svgEx.attr('width', windowWidth);
svgEx.attr('height',windowHeight);
svgEx.attr('viewBox', viewBox);

let zoomArea = svgEx.append("g").attr("id","zoomArea")
.attr("transform", `translate(0,${taskHeight + margin.top + margin.bottom})`)
    ;

let gridArea = svgEx.append("g").attr("id","gridArea")
.attr("transform", `translate(${controlAreaWidth + margin.left},${margin.top
    })`);

let controlArea = svgEx.append("g").attr("id","controlArea")
```

```
.attr("transform", "translate(0,0)");

let poolArea = svgEx.append("g").attr("id","poolArea")
.attr("transform", `translate(${controlAreaWidth +  taskWidth + margin.top + margin.bottom},0)`);
```

Listing 4.2: Creation of groups

The result is a division such as of Figure 2.

Figure 2: Division of the areas.

4.2.2 Graph and Grid

The first building block is the grid. Thank to D3 functions it's possible to build one with ease.

The first thing to do is creating a scale for both axes. According to the kind of graph there are multiple possibilities. The most common are time axes and band axes, or axes whose terms are a list of items, for instance teams names or courses categories. Here I will show a version that includes both, a time axes in the X-axes and a band axes in the Y-axes.

Initial declaration and initialization of variables is needed. An empty array for tasks and pool tasks, a Set for the teams and a minimum and maximum date, set respectively to the current date and 100 days in the future. Then the team set is populated.

```
let tasks = [];
let poolTasks = [];
let teams = new Set();
teams.add("Team A");
teams.add("Team B");
teams.add("Team C");
teams.add("Team D");

let minDate = new Date();
let maxDate = minDate.addDays(100);
```

Listing 4.3: Decaltarion and Initialization

Subsequently, it's possible to define a scale, an object with two attributes containing, respectively, a scale for each axes. Each scale must have a domain, the logical space, and a rage, the physical space in the layout. The domain for the x-axis is from the minimum date to the maximum date, and for the y-axis it's the set of teams. The range is the physical area of extent that the axis occupy, for x-axis from 0 (this will be appended to the grid area, thus, 0 is the beginning of the task area without the margins) to the width of the task area, and for y-axis from the top if the grid area height to the bottom 0. Some graphical adjustments are made to make ticks wide and not continuous (dash array), and to make custom tick formats for date. Then, the axis are moved to the desired positions through a "transform" attribute.

```
let scale = {
x: d3.scaleTime()
.domain( [minDate, maxDate] )
.range([0, taskWidth]),
y: d3.scaleBand()
.domain(Array.from(teams))
.range([taskHeight, 0])
}

let xAxis = d3.axisBottom(scale.x).tickFormat(d3.timeFormat("%b %d"));
let yAxis = d3.axisLeft(scale.y).tickSize(-taskWidth - 10).tickPadding(0);

gridArea.append("g").attr("class", "X-axis").attr("id","X-axis")
.attr("transform", `translate(0,${taskHeight})`) //translate to bottom
.call(xAxis);
```

```
gridArea.append("g").attr("class", "Y-axis").attr("id","Y-axis")
.call(yAxis);

gridArea.select("#Y-axis").selectAll(".tick line")
.attr("stroke", "#777")
.attr("stroke-dasharray", "2,2")
.attr("transform", `translate(0,${ taskHeight / teams.size / 2 })`);

gridArea.select("#Y-axis").selectAll("text")
.attr("text-anchor", "middle")
.attr("transform", `translate(-${margin.left / 2.5}, -0) rotate(-90)`);
```

Listing 4.4: Creation of groups

The result is shown in figure Figure 3

4.2.3 Virtual Sticky Notes

This passage is the most subject to changes and variations according to the kind of data to display and the information contained in the virtual sticky notes. This translates in attributes that can largely vary according to the study case, so I'll present a general approach to represent a sample data into a virtual sticky note.

My sample task is an object that has the following attributes: a name, a start date, a duration (in days), a team. The following code shows the creation of one example:

```
let task = {
name: "Task 1",
```

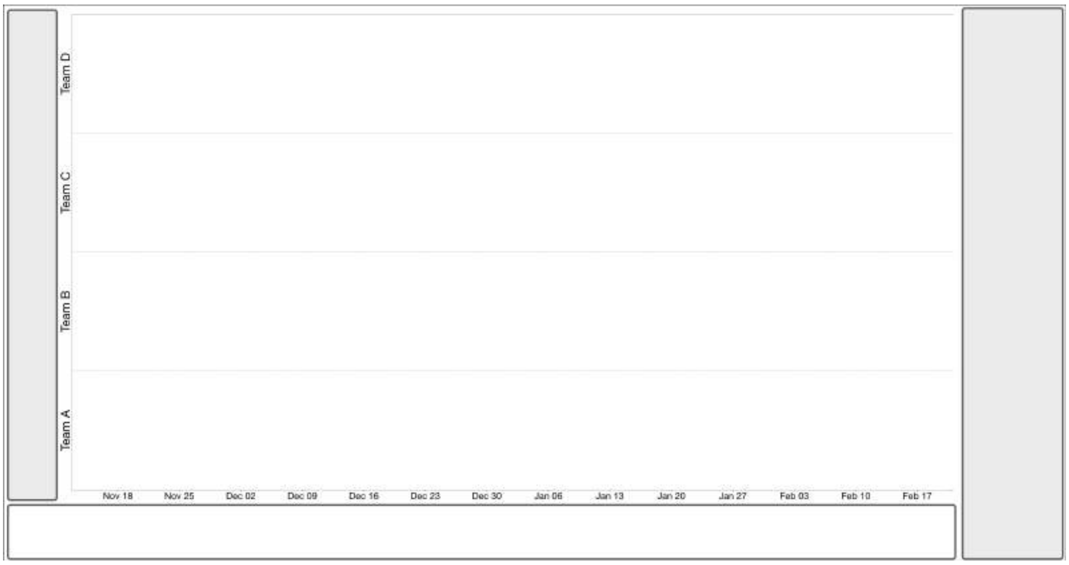

Figure 3: Representation of the Grid

```
start: new Date("01/16/18"),
duration: 5,
team: "Team A",
}
tasks.push(task);
```

Listing 4.5: Populating tasks with a sample task

The Real case tasks present a more detailed structure with more attributes containing different sorts of information. Caterpillar tasks contain 11 attributes, the ones for project planning course just 7.

Then, it's possible to start the creation of the visual representation of such sample task. First, creating a SVG group that will contain all the tasks. Each task is represented by a rectangle and a text that will be contained in another group.

In order to associate data to SVG elements the D3 library offers handy mechanisms. What we will need is to access the "enter selection", all the data that has not already been assigned to an SVG object.

```
let totalDays = (maxDate - minDate) / 1000 / 60 / 60 / 24;

let taskGroup = gridArea.append("g").attr("id", "Tasks");
let enterSelection = taskGroup.selectAll("g.task").data(tasks, (d) => [d.name
    ,d.start]).enter();
let t = enterSelection.append("g").attr("class", "task");

t.append("rect")
.attr("x", function(d) { return scale.x(d.start); })
.attr("y", function(d) { return scale.y(d.team) ; })
.attr("width", function(d) {return +(taskWidth * d.duration / totalDays) ;})
.attr("height", scale.y.bandwidth())

t.append("text")
.attr("text-anchor","middle")
.attr("dy", "0em")
.text(function(d) { return d.name })
```

```
.attr("x", function(d) { return scale.x(d.start) + +(taskWidth * d.duration /
    totalDays) / 2; })
.attr("y", function(d) { return scale.y(d.team) + scale.y.bandwidth() / 2 ;
   })
```

Listing 4.6: Creating the graphical element

It's possible to customize some graphical aspects of the rectangle: setting round edges, fill color, stoke color and width.

```
.attr("rx", 5)
.attr("ry", 5)
.attr("fill", "blue" )
.attr("stroke", "black")
.attr("stroke-width", strokeW )
```

Listing 4.7: Customizing Rect

Similarly, it's possible to customize some graphical aspects of the text as well: setting the font, the font size, stoke color and width, and fill.

```
.attr("font-family", "Helvetica Neue")
.attr("font-size", fontSize)
.attr("fill","white")
.attr("stroke","none")
.attr("stroke-width","0")
```

Listing 4.8: Customizing text

In the piece of code above some general variable have been used just as representation purposes.

The result is shown in Figure 4

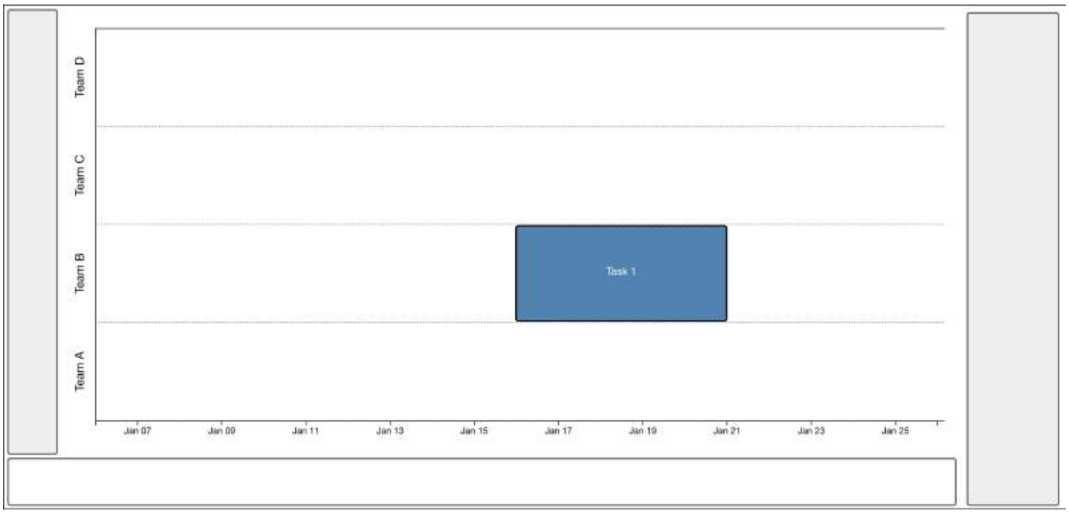

Figure 4: Graphical representation of a Sample Task

4.2.4 Buttons

Buttons have an easy structure, they are made of an SVG group containing rectangle and a text to which it has assigned a routing on click.

First, definition of buttons dimensions and number is needed: width that is a little bit smaller that the area they are assigned, a fixed x coordinate for each button, and a variable y, since they are built in vertical. Also, I provide a simple example of how measures can be defined at run time depending on screen size.

```
let buttonNumber = 4,
buttonDiv = buttonNumber + 2;
let buttonW = controlAreaWidth * 0.8,
buttonH = controlAreaHeight / buttonDiv,
buttonX = (controlAreaWidth - buttonW) / 2,
buttonY = [];

for (let i=0; i<buttonNumber; i++){
buttonY.push((controlAreaHeight / buttonNumber - buttonH) / 2 + ((
    controlAreaHeight) / buttonNumber) * i) ;
}
let round = Math.round((windowWidth + windowHeight) / 700),
fontSize = Math.round((windowWidth + windowHeight)/ 130),
strokeW = Math.round((windowWidth + windowHeight) / 750);
```

Listing 4.9: Buttons parameters definition

Then, it's possible to create groups for buttons to append directly to the control area group. Following, an example of the creation of the first button with some graphical adjustments.

```
controlArea.append("g").attr("id", "button1")
.on("click", handleClickRoutine1)
.append("rect")
.attr("x", buttonX)
.attr("y", buttonY[0])
```

```
.attr("width", buttonW)
.attr("height", buttonH)
.attr("rx", round)
.attr("ry", round)
.attr("fill", "#009688" )
.attr("stroke", "#004D40")
.attr("stroke-width", strokeW)

controlArea.select("#button1").append("text")
.attr("font-family", "Helvetica Neue")
.attr("font-size", fontSize)
.attr("fill", "white")
.attr("text-anchor", "middle")
.text("Button 1")
.attr("x", buttonX + buttonW/2)
.attr("y",  buttonY[0] + buttonH/2)
.attr("transform", `rotate(-90 ${buttonX + buttonW/2} ${buttonY[0] + buttonH
   /2})`);

function handleClickRoutine1() {
}
```

Listing 4.10: Buttons definition

To allow more feedback to the user, it's possible to define routines for hovering with mouse on the button highlighting it adding the following to the button definition:

```
.on("mouseover", handleMouseOver)
.on("mouseout", handleMouseOut);
```

Listing 4.11: Buttons hovering

And defining simple hovering methods. With D3 it's possible to select the SVG object and change its attributes. This methods can be applied to the task boxes previously described.

```
function handleMouseOver () {
d3.select(this).attr("fill-opacity","0.33").attr("stroke-opacity","0.75");
d3.select(this).selectAll("text").attr("fill","#434343").attr("fill-opacity",
    "1")
}
function handleMouseOut () {
d3.select(this).attr("fill-opacity","1").attr("stroke-opacity","1");
d3.select(this).selectAll("text").attr("fill","white")
}
```

Listing 4.12: Hovering functions

An example with 4 buttons is shown in Figure 5

4.2.5 Pool Area

The pool area is for tasks that do not have enough information on the attributes necessary to place them in the grid, in this example either a start day, a duration or a team. The procedure to create a visual representation of such tasks is quite similar to the creation of regular task. The differences are in the dimensions of the rectangle and the position.

Figure 5: Layout of 4 buttons while hovering on Button 3

Height and position of drawn tasks can be adjusted to make space for virtually infinite tasks. In the following example space is divided to make room for 8 tasks, and when more shrinking happens.

```
let height;

if (poolTasks.length <= 8) height = (poolHeight - 20) / 9;
else height = (poolHeight - 20) / (poolTasks.length + 1);

let poolEnterSelection = poolArea.selectAll("g.poolTask").data(poolTasks, (d)
    => [d.name,d.start]).enter()
```

```
let newGroupPoolTasks = poolEnterSelection.append("g").attr("class", "
    poolTask")

newGroupPoolTasks.append("rect")
.attr("x", margin)
.attr("y", margin + height * i)
.attr("width", poolWidth - margin )
.attr("height", () => height)

newGroupPoolTasks.append("text")
.attr("text-anchor","middle")
.text((d) => d.name)
.attr("x", () => poolWidth / 2)
.attr("y", (d,i) => (margin + height * i) + height/2);
```

Listing 4.13: Pool Area Creation and population

In Figure 6 two examples of the Pool Area behavior, with few tasks and many tasks.

4.2.6 Zooming and global view

For the zooming functionality we need two different parts: a brush and the global view. First, we build the global view: there is the need to create a new scale, that will remain fixed, and use it to modify the general scale.

```
zoomScale = {
x: d3.scaleTime()
.domain([minDate, maxDate])
```

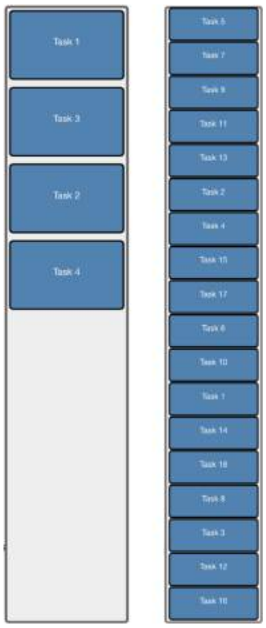

Figure 6: An example of Pool Area with 4 tasks on the left, and an example with 18 tasks on the right

```
.range([0, zoomTaskWidth]),
y: d3.scaleBand()
.domain(Array.from(teams))
.range([zoomTaskHeight, 0])
}
let zoomGroup = zoomTaskArea.append("g").attr("id", "TasksZoomBar");
let t = zoomGroup.selectAll("rect").data(tasks, (d) => [d.name,d.start]).
    enter();

t.append("rect")
.attr("x", function(d) { return zoomScale.x(d.start); })
```

```
.attr("y", function(d) { return zoomScale.y(d.team) + 1; })
.attr("width", function(d) {return +(zoomTaskWidth * d.duration / totalDays)
    ;})
.attr("height", zoomScale.y.bandwidth())
```

Listing 4.14: Zoom Area Global View Creation

Subsequently, I need to draw a brush, that will be used to zoom.

```
let brushSelecion = [0, zoomAreaWidth],
brushing = false,
dragW = Math.round((windowWidth + windowHeight) / 150);

let brush = zoomArea.append("g").attr("id", "brush");

brush.append("rect").attr("id", "selector")
.attr("x", 0)
.attr("y", 0)
.attr("width", zoomAreaWidth)
.attr("height", zoomAreaHeight)
.attr("fill-opacity","0.25")

brush.append("rect").attr("id", "brushL")
.attr("x", -dragW )
.attr("y", -5 )
.attr("width", dragW)
.attr("height", zoomAreaHeight + 10 )
.attr("fill", "none")
```

```
brush.append("rect").attr("id", "brushR")
.attr("x", zoomAreaWidth - dragW)
.attr("y", -5)
.attr("width", 2 * dragW)
.attr("height", zoomAreaHeight + 10)
.attr("fill", "none")
```

Listing 4.15: Zoom Area Brush Creation

I won't report the definition of the routines when dragging one of the 3 selectors because form this point the code becomes dependent on the SAGE2 platform. D3 has brushing functions already implemented, while, due to the impossibility of using regular mouse actions, I had to create a custom version of such functions. The logic behind this is to remap the general scale based on the selection extension and calculate a "k" zooming factor to apply to some parameters of the original scale and all the rectangles of the tasks in the grid. Since we want a zoom only on the x-axis. Once these two actions have been performed we can perform the following operations when the brushing event occurs:

```
app.gridArea.select("g#Tasks").selectAll("g.task").selectAll("rect").attr("
    width", (d)  => (taskWidth * d.duration / totalDays) * k )
.attr("x", (d) =>  scale.x(d.start) );
app.gridArea.select("g#Tasks").selectAll("g.task").selectAll("text").attr("x"
    , (d) => scale.x(d.start) + (taskWidth * d.duration / totalDays) * k / 2);
```

Listing 4.16: Zoom Area Applying the zoom

As mentioned before, this functionality has been omitted in the study course planning application due to the nature of the time axes and grid item dimensions. But it's generally considered a useful functionality when creating a project planning application that deals with timed tasks.

After populating the grid with more tasks and adding some more graphical enhancements, such as more colors and shapes, and after having added some minor graphical changes in the brush to make it more visible, the result is shown in Figure 7. The zoom in action is shown in Figure 8.

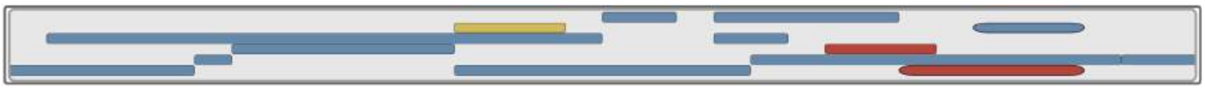

Figure 7: Zoom Area with global view and brush. The grid has been populated for illustration purposes

4.3 Study Course Planning Application

This section shows the study course planning application I realized for the user study. Users will be able to create courses, enrich them with information about days, times, and credit hours, move them, and modify them. The application has been developed in a period of 2 terms in the EVL at UIC, where several wall displays are installed and running SAGE2.

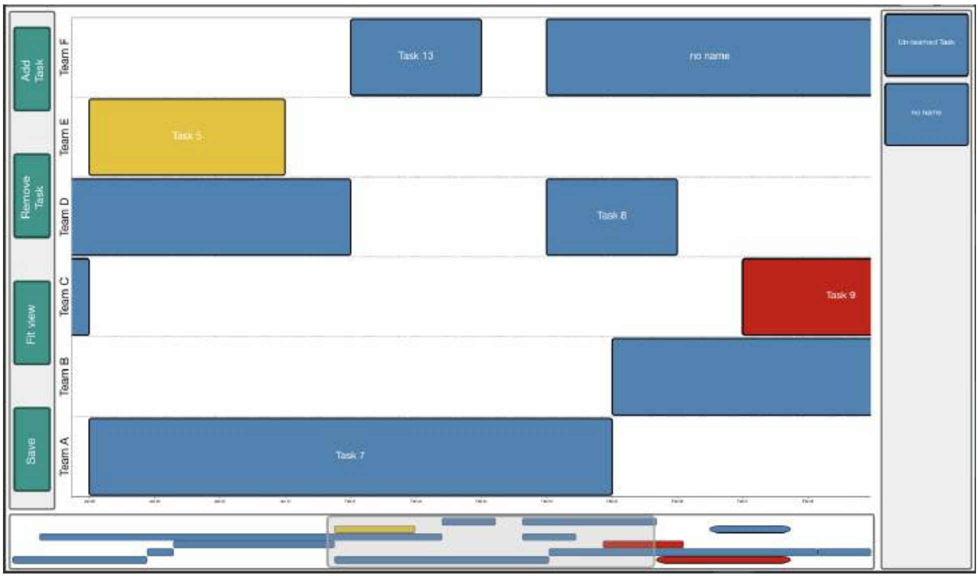

Figure 8: Zoom action: the brush selects the zoomed view that appears in the main grid

This application offers all the functionalities that users need to create a plan for a study course. It allows co-located collaboration, and allow users to manage the scheduling progress. Users will visualize a grid representing the time and course categories. Users will add courses one by one and place them into the rid to reach the goal of creating an admissible course plan.

4.3.1 Application Initialization

Before showing the actual interface I will explain how the application is initialized. The initialization can follow 2 slightly different procedures.

The first is when the application is regularly launched on the SAGE2 environment. It is possible to do so by uploading the compressed file in .zip extension on the SAGE2 server using the web interface. In this case the application will create the basic data structures to handle

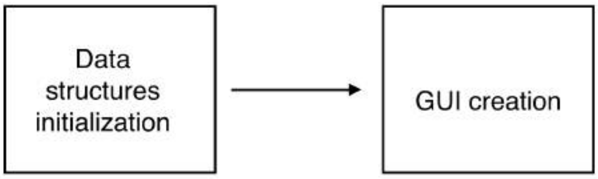

Figure 9: The process of initialization in case no configuration file is detected

later courses addition and set the time boundaries for the grid. Also it defines the course category boundaries, another element of the grid. A collision table that will help to determine how many courses there are in the same cell, is created and initialized. Subsequently, it looks for a configuration file in CSV format to parse. If not present the GUI will start to be created.

If the file exists, the second procedure is started and before creating the GUI this file is parsed, as shown in Figure 10. The data structures previously created are filled, and the GUI is created showing all the parsed items. Some filtering on the data read is performed to correct the most common errors and mistakes on the file. All the courses that cannot be assigned a term or a course category end up in the Pool lane, an portion of the screen created for storage of uncategorized items. Each course have the following fields:

- name: the name of the course. This is the identifier key for courses.
- category: either one of Computer Science, Design, General Elective. This will be displayed as the Y-axes.

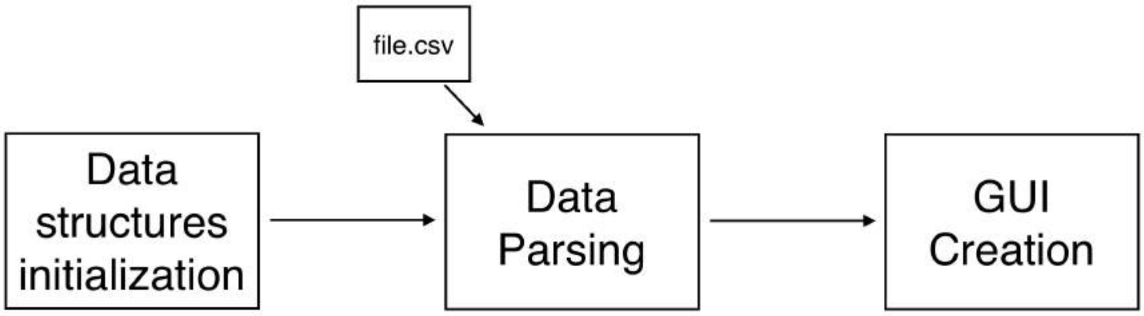

Figure 10: The process of initialization in case the configuration file is detected

- term: the term in which the course will be held. This will be displayed as the X-axes.
- days: the days in which the course takes place.
- hours: the hours within the days.
- credit hours: the credit hours recognized for enrolling in the course.

The GUI creation starts with the drawing of the buttons area, proceeds with the drawing of the grid area, and finally it displays the Pool area. In the general version of the project planning application there is also a fourth region dedicated to the zoom functions. The majority of the space is covered by the grid.

4.3.2 Interface and Design

In this subsection I will illustrate, with the help of some images, taken on the web display interface of SAGE2, how the interface of the course scheduling application is composed.

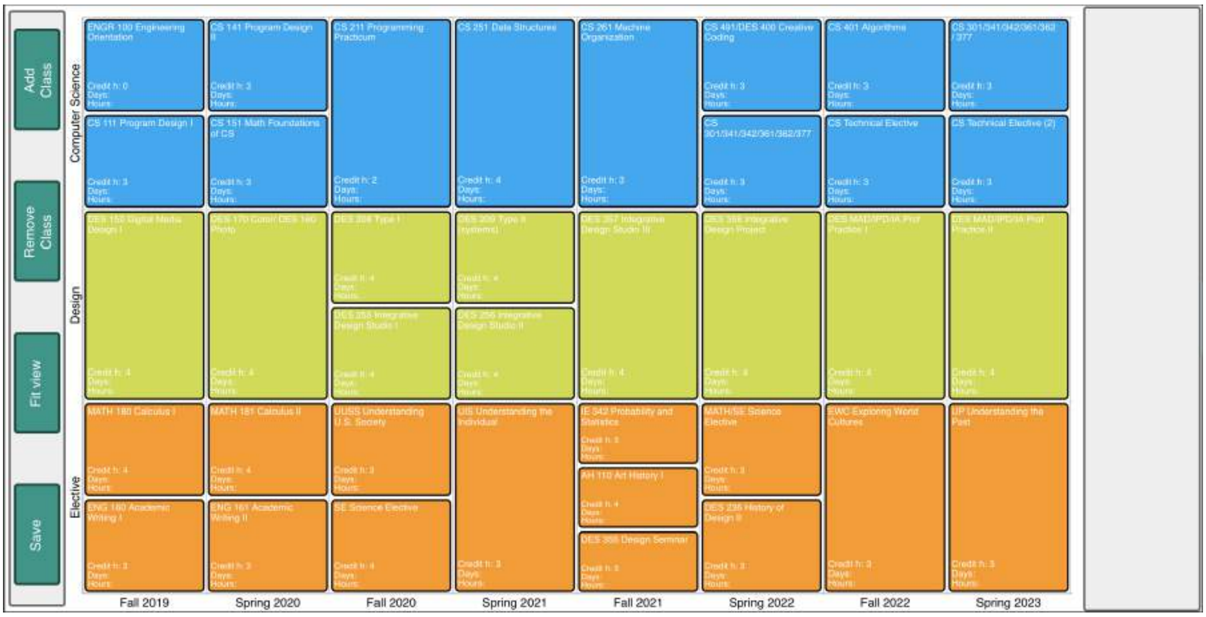

Figure 11: The course plan interface filled with some courses for illustration reasons. The center is occupied by the principal grid filled with courses. On the left the buttons area, and on the right the Pool lane.

In figure Figure 11 the application layout is presented. In the center the principal grid is covering the most of the space, to have the most space occupied with the most important pieces of information: the courses. The X-axis is determined by the terms of the study course, that is composed of 4 academic years, for a total of 8 terms. The Y-axis is composed of the courses category, that can be either "Computer Science", "Design", or "General Elective".

Text sizes and line thicknesses are always determined at run-time, allowing them to adjust themselves to smaller and larger displays dynamically. The technique comes into action also when resizing the application window in SAGE2.

Every course is determined by a virtual sticky-note, or a box, in the grid. If only one course, in a specific term and of a specific category, is present, it will tend to occupy all the spaces possible. If more than one is present they will occupy the half of space each, and the more they are the less space they will occupy. The correct positioning of the "conflicting" courses is managed by an external structure that is keeping track of how many courses there are in each cell. The text size is also reduced depending on how many courses are there in a specific cell, the more courses, the smaller the text size, to fit all the text in a smaller area.

All information on a course is displayed in each box. The course name at is positioned at the top and its text length is controlled by an algorithm. Unfortunately there is no ready formula to fit text in a box in javascript nor D3, so I had to create a personalized one.

The text fitting algorithm

The algorithm I created dynamically calculates the space, in pixels, occupied by the course name according to character size and type (for example the same word in "Times new Roman" occupies different space than in "Helvetica") and compares it with the space that the its box offers. If the text is too long to fit in a single line it will automatically truncate the name and create a second line, or how ever many are necessary to make room for all the title. If it is still too large, filling all the box, the algorithm repeats the procedure again shrinking the size. If the text is stil too large it keeps shrinking, but if the text size becomes too small, passing the readability threshold, the algorithm performs truncation of the name, starting from the end until it's possible to fit in the box. This algorithm was developed to face the problem of very long names and very small boxes for the general project planning version. For the study course planning version this problem should not occur since the course names are generally not long and boxes have plenty of space. Thus, the shrinking and truncating stages may not be activated.

The other relevant information is displayed on the bottom of the box, one per line.

On the left there is the button area, text is rotated so the buttons occupy the least amount of space. There are 4 buttons, each one with a dedicated function: "Add Course" it allows users to add a new course, "Remove Course" it allows users to delete an existing course, "Fit View" acts like a home button, and "Save" allow users to export the current session in a csv file.

On the right there is an area called the "Pool Area", where all the tasks not assigned to a term or a category end up. The courses here show the same information as the ones in the grid. In section 4.3.3 will be explained how courses can be moved to the grid and modified.

4.3.3 Function and Features

Given the description of the interface in the previous section, I will now describe functions and features, including all actions users can perform.

Interaction with the application uses a mouse and keyboard. The user interacts not with a regular mouse pointer, but the one provided with SAGE2. Indeed, to connect to the application users must connect to the SAGE2 web interface, through a regular web browser and connecting to the SAGE2 address for the particular wall. In the web interface it's possible to upload a csv file, that will be automatically opened with the course planning application, and it will trigger the launch of the application itself. Or it is possible to launch the application with no input

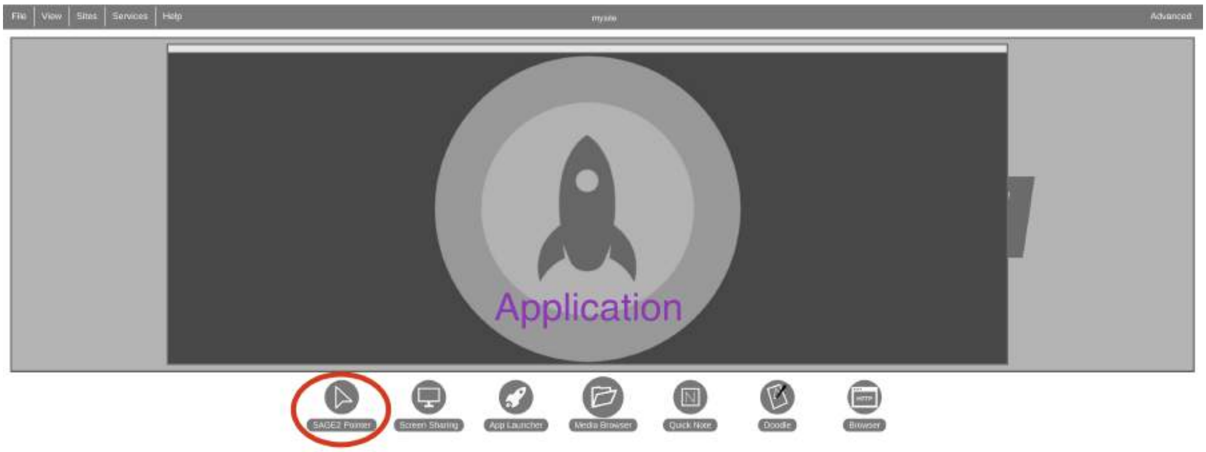

Figure 12: The input client interface, open in a regular web browser, generally on a laptop. There is an open application on the SAGE2 wall, and the button to access the SAGE2 pointer to interact with the application is highlighted.

file, this will result in a blank grid, in order to create a new plan from scratch. At this stage, clicking to the SAGE2 mouse button will provide access to the SAGE2 mouse in the display client, generally open in the wall display, with the input client, generally open in a regular laptop. In Figure 12 there is shown a common use case of the SAGE2 input client, with the open application, and button highlighted to access the SAGE2 pointer.

When the user has access to the SAGE2 pointer, interaction with the application can begin. I will present all the possible interaction dividing them from areas.

4.3.3.1 Button Area

In the button area there are four possible interactions, one per button. Buttons will highlight when hovering with the SAGE2 pointer to make users have a more natural interaction.

Add Course

Clicking the "Add Course" button will open a new pop-up window where a user can insert all the necessary fields to create a new course.

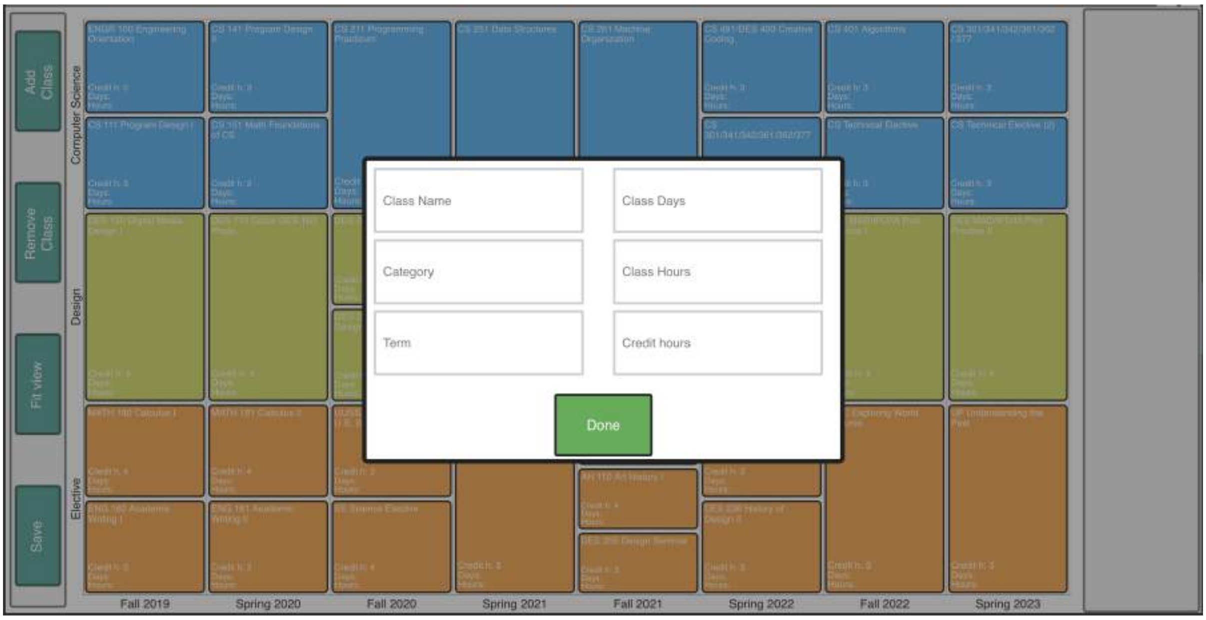

Figure 13: The scenario after clicking on the button "Add Class". Six fields can be filled with information about the course to be added.

As the Figure 13 shows, in the new window six fields appear, and users can decide to fill just some of them. Filling the "Category" and "Term" fields will create a new class that will be placed in the grid. If one of them is left blank or filled with incorrect or non recognized input data, the new class will be placed in the Pool area. Some input correction is being performed while filling fields. For example, leaving the "Name" field blank will assign the "no name" string to the name attribute of the course. Also, some shortcuts are performed to let users write new classes faster, for instance when filling the "Term" field, it's not necessary to write the full word but just the initial of the term and the last 2 digits for the year, so instead of writing "Fall 2019" a user could simply write "f19" and it will be translated to "Fall 2019". Similarly for categories, "cs" will translate to "Computer Science" and in a analogous way for all the other categories and terms.

An early version of the application used the standard "prompt" embedded in Javascript, but the regular browser pop-up windows could not be activated in SAGE2 environment, so I needed to develop an ad-hoc solution. This is the only part where some languages other than Javascript has been used. Indeed, to create the layout for the input fields I used HTML forms, to save the input information and send it to the application.

Remove Course

Clicking the "Remove Course" button will enter the delete mode. At this point the user who clicked on the button can click on any course, whether in the grid or the pool area. Then a confirmation window will appear, as shown in Figure 14

Figure 14: The scenario after clicking on the button ”Remove Class”. After clicking on a course, a confirmation window will appear.

Other Buttons

The third button, ”Fit View” acts as a home button, bringing the view to the original global view. It was created for the general project planning version and thought to be a shortcut to zoom back. In the study course planning software its functionality is less relevant.

The fourth button, ”Save”, allows to save the current session. It generates a new csv file in the SAGE2 folder that is possible to download for further use. To restore the session it sufficient to open the file directly from the SAGE2 folder.

4.3.3.2 Grid Area

In this area the main interactions are with classes. If no classes are present then no interaction is possible within this area.

Move

When some classes are created, or loaded from a file, its possible to change their two most important attributes, i.e. term and category, with a simple gesture. Clicking and dragging a class will cause it to be moved. While moving the class, the virtual sticky-note, or the box representing it, will not move freely but I developed an algorithm that will snap it to the grid. In this case a 3x8 grid. Releasing the mouse click will cause the change to actually take place, modifying the data of the course related to box. In D3 it is possible to link graphical elements to data, and this allowed me to find data related to a specific box and vice versa.

While dragging, if the pointer is brought outside the boundaries of the grid, it will start moving freely. If the mouse release happens at this point, the class will be moved to the pool area. The box in the grid will disappear and it will appear in the corresponding one in the pool lane.

Modify

A click on a task will cause a new pop-up window to open. Such window have similarities with the one presented in section 4.3.3.1. The window position will be dependent on the class box position and it occupies a restricted area, so not to conflict with other users work. If the resulting position is too close to the boundaries of the application, causing the window to be created outside the visible area, the window will be shifted toward the center just enough to make it completely visible.

The window has the same six fields, "Name", "Category", "Term", "Days, "Hours", "Credit hours", as shown in Figure 15, and they work in a slightly different way. To make the modification as fast as possible, the same logic explained in section 4.3.3.1 for quick entries for "Category" and "Term" is applied and an attribute of the data will be modified only if the filed in the window is filled, otherwise it will remain untouched.

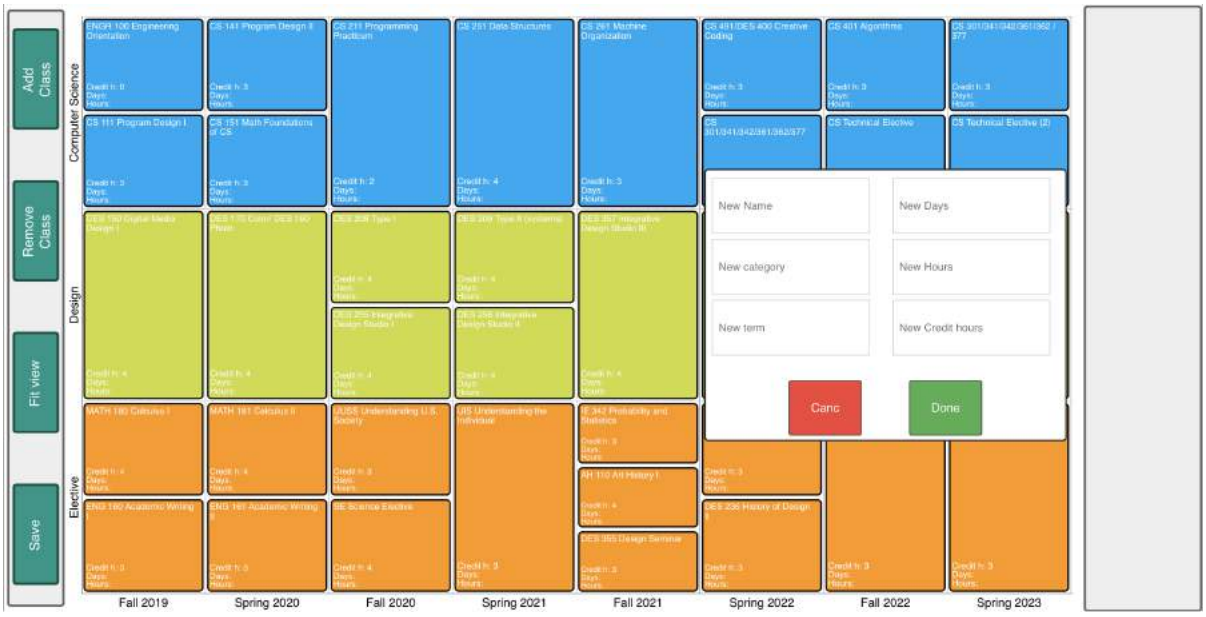

Figure 15: The scenario after clicking on task. The pop-up window allows access, through D3 functions, to modify the class fields. Other users are still able to work on other courses.

4.3.3.3 Pool Area

This area offers very similar interaction to the Grid Area and is described with detail in the previous paragraph.

Clicking will open the same pop-up window that allows users to modify the class fields, and dragging tasks will allow users to move them. If kept outside the grid they will move freely, and if released in this region the box will return to its previous position and the class data will be unmodified. If the dragging reaches the grid, an adapted version of the snapping to grid algorithm is applied so the box will snap to the grid. If released here, the data is updated along with the grid and the pool area view.

The pool area, as it is designed, can fit at most 8 classes. If more classed are dropped here an algorithm will manage the shrinking of all of the classes to make room for the new entry. This allows the area to fit virtually infinite classes.

4.3.4 Architecture

In order to understand how the previously described course planning application works, we need to take into account other components, the SAGE2 components.

- SAGE2 Server
- Input Client
- Display Client

My application is built on top of SAGE2, running on the SAGE2 server. Every time an interaction is performed the input client sends the communication to the server, that forwards it to the application. The latter communicates the changes to the server that provides the information changes to the display client. A general interaction workflow is shown in Figure 16, and is explained with more detail in the next paragraph.

Interaction Workflow

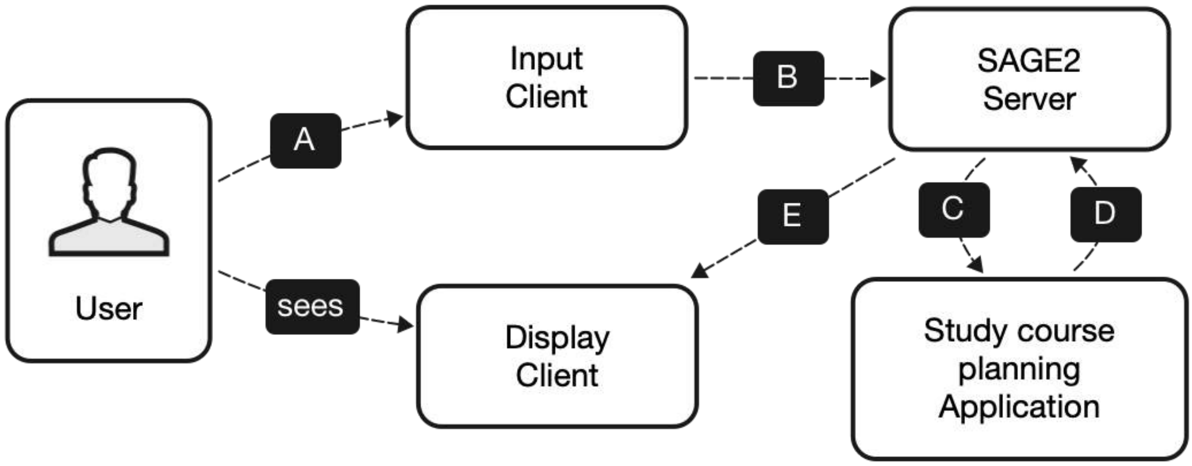

Figure 16: The workflow of a general interaction.

The interaction workflow starts with a user interacting with the input client (A), opened in his or her web browser, with the intention to click or drag something on the application's graphical interface. At the same time the user is looking at the display client. When the interaction occurs, the message is sent to the server (B), that performs notification to the application that is running on the server (C). The application then applies the changes, if any, and communicates them back to the server (D). Then the server needs to update what the user sees, i.e. the display client (E). At this point the user finally has visual feedback of his or her initial interaction.

4.3.5 Model

The application model shows similarities with the common paradigm MVC, or model view controller. As shown in Figure 17, we can decompose the course planning application in 3 different components that resemble the ones of the MVC. The Model is represented by the the "Data Model", the GUI represents the view and the functions dedicated to the input response are the controllers.

The user, at first, needs to acquire information through the GUI. Then, through mouse and keyboard, he or she provides input to the system, such as click, drag, type, or even just moving the pointer. Dedicated procedures and functions handle the input received and if necessary update the model providing it with the respective changes. The model then updates the GUI, that notifies the user of the changes made.

4.4 Variations and Possibilities

The course planning application explained in detail in Section 4.3 is just one implementation of the techniques presented in Section 4.2.

During the development of this specific version, I created some different variations, that I will show here, and they will function as examples and inspirations that can help to unlock ideas for new possibilities, and the creation of more and different versions for other purposes.

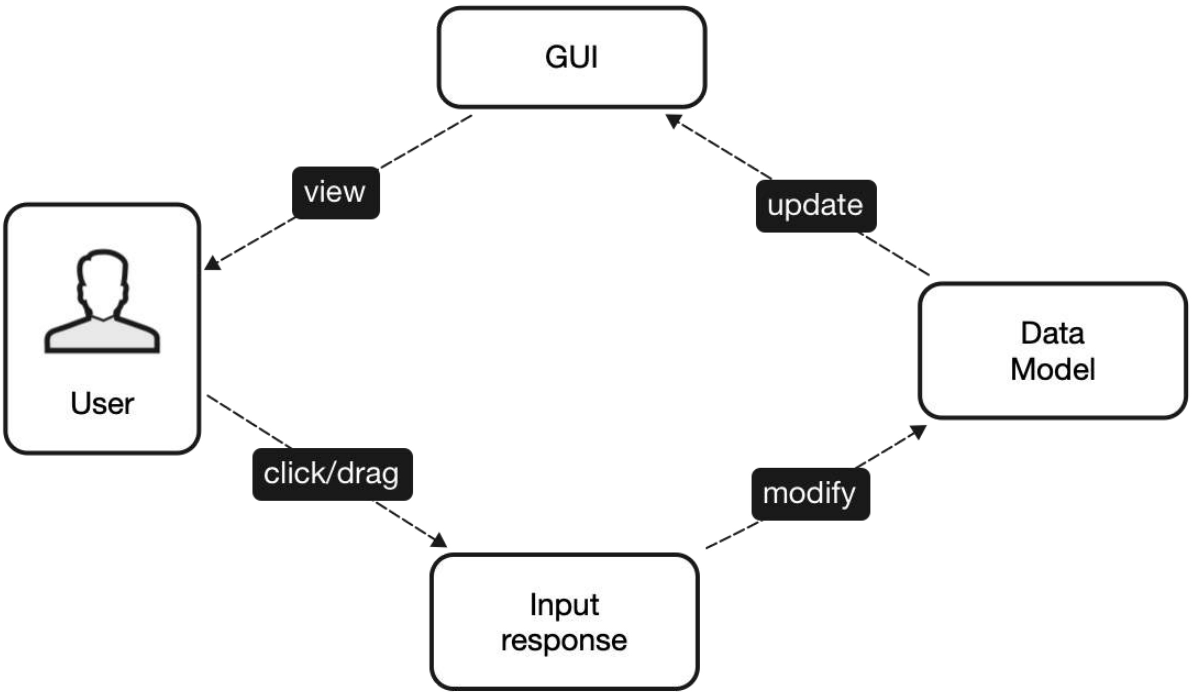

Figure 17: High level representation of the study course planning application.

4.4.1 Project planning application

This is the most general version, that is the most flexible and feature complete. This version allows users to manage a project plan of any duration, from days to years-long.

It's possible to notice that tasks automatically adjust their length based on their duration. The X-axis is composed of a time-line that auto-adjusts itself based on the data read on the input file. The Y-axis is composed of as many lanes as different teams or resources that are needed. In this case the grid resolution on the horizontal side is very tiny and a "cell" occupies just few pixels since the minimum step is one day. On the other hand, vertically the step is a team lane.

The first example in Figure 18 shows a general example, and Figure 19 shows its usage in a real use case scenario. The data comes from a real conference, though the text descriptions have been replaced. As the reader can see, the amount of tasks is enormous, and can be even larger, thus the zoom function in such scenario comes very handy. In the general example color coding has been used to mark task priority. "Blue" for low priority, "yellow" for medium, and "red" for high priority. Additionally, tasks can be marked as "milestones" and this will provide different visual feedback in the grid. Milestones present a more round shape and have thicker borders. In the second example, for the real case scenario, the milestone feature was not required and color coding had a different meaning: every team had been assigned a different color. In fact, the Y-axis in this case was represented by allocable resources.

This is an example of how different the requirements and functionalities required can be in different real cases, which is one of the reasons why people still prefer the traditional approach, as explained with more detail in Section 2.4.1.

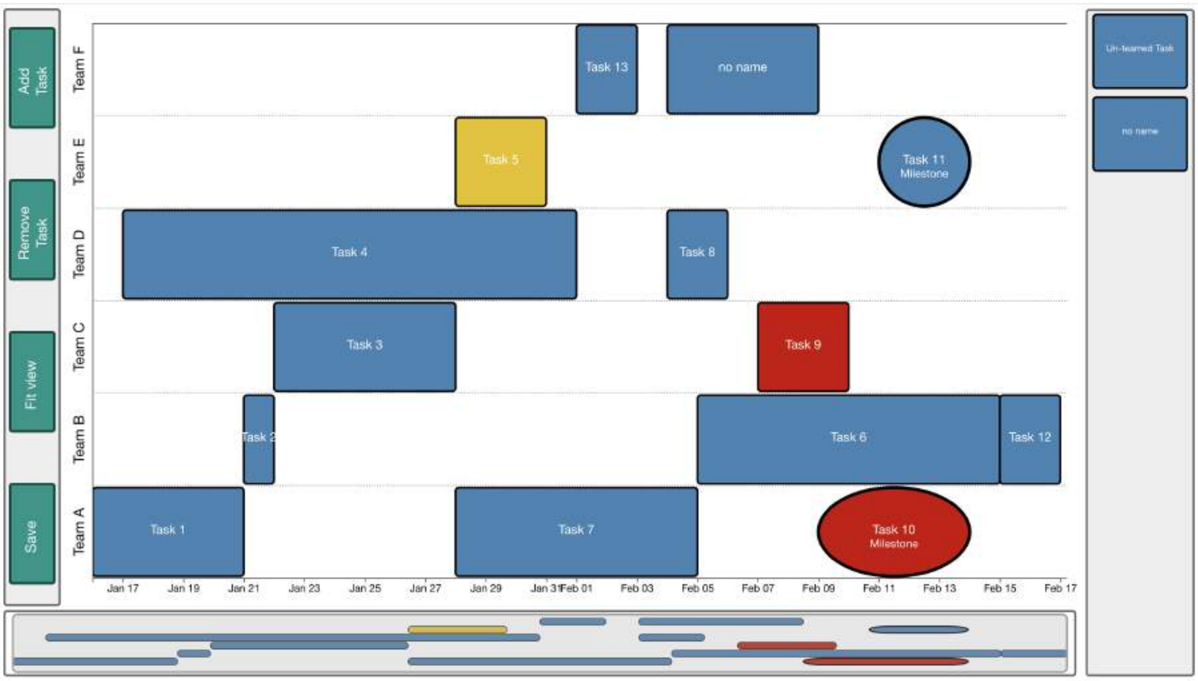

Figure 18: Variation 1: Project planning application. The current view is showing some sample tasks assigned to some sample teams. Milestones have been marked differently, and the zoombar at the bottom is showing the global view. Some uncategorized tasks are placed in the pool lane.

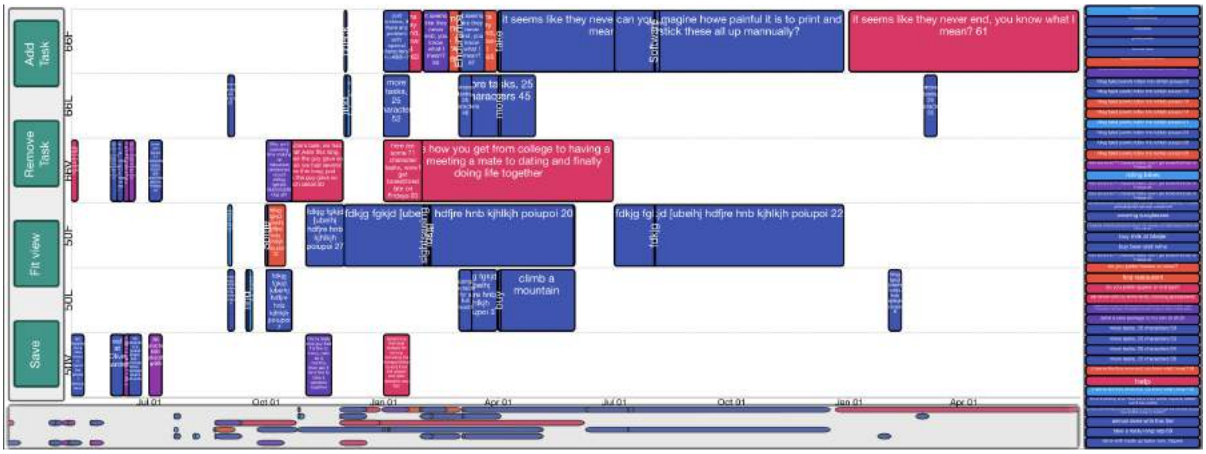

Figure 19: The Project planning application showing the data of a real use case conference. Task number is over a hundred.

4.4.2 Courses scheduler application

This version is thought to visualize, but also create and modify, a schedule for a term of study of courses. In Figure 20 a full schedule for a semester of all the courses of the computer science department is shown. The courses are taken from real ones of the course offer of UIC CS department.

The grid is divided per days of the week, on the X-axis, and hours of the day, on the Y-axis. The functionalities are similar to the study course planning application, as described in detail in Section 4.3.3.

A good idea for a further development is to integrate both views in the same application. This view indeed could be the detailed representation of a specific term in the larger view of the course planning application. With an interaction gesture, e.g. a double click in the area, or a specific button press, the view could be brought from the general eight terms view to the more detailed view for the selected semester.

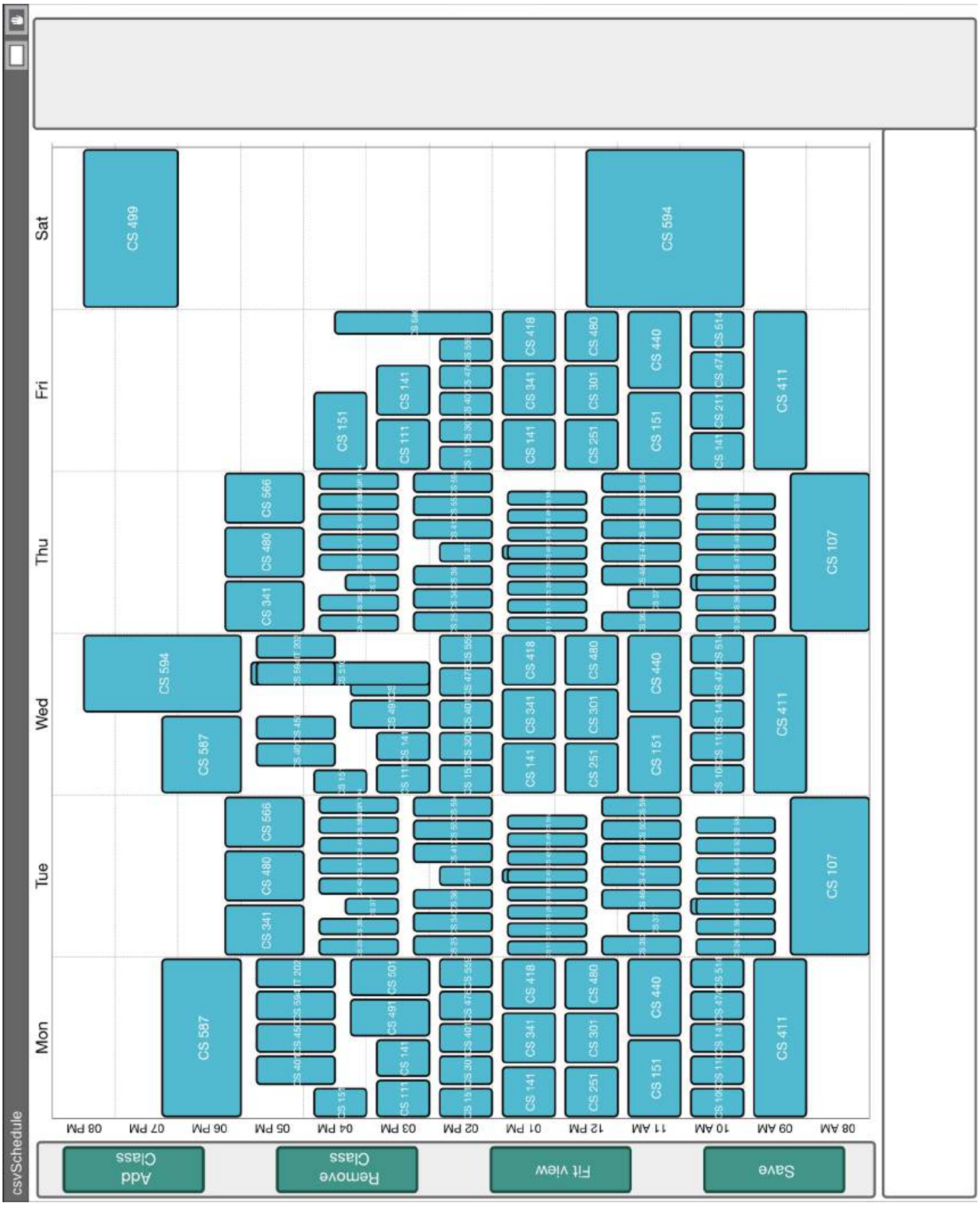

Figure 20: Variation 2: Courses scheduler application. The current view is showing a full term course offer of the computer science department of UIC

CHAPTER 5

USER STUDY

In chapters 2 and 3 are shown the advantages and benefits that large format displays provides in several different applications, from solving complex tasks that require high cognitive loads, to navigation of 3D environments. On the other hand some drawbacks and challenges have been shown. There is no paradigm to follow while creating an application for large format displays and sometimes their use can be less intuitive than their regular sized counterparts.

Following, studies on collaboration and collaboration using a large displays were discussed. As result LVDs are shown to have an advantage in collaborative tasks and help productivity in many different settings and topics. Examples discussed in previous chapters include elaboration of large amount of text documents, and modeling of a business process. Although, no studies were conducted on large format displays in the field of project planning.

In this research we are interested in studying the advantages and improvements that large format displays can bring to a collaborative environment in the specific field of project planning. Tasks on this field require the processing, categorization and collocation of a large amount of information, while respecting constraints of several different typologies.

The nature of data collected on this study make the analysis conducted of the "qualitative data analysis" kind. Such an approach is necessary when collected data is non-numerical, and features and patterns relevant to the research object must be retrieved from the qualitative data, that in our case consists of audio and video recordings. Some background on this topic is provided in section 2.5

5.1 Motivation

The purpose of this study is, not only to confirm previous studies' results that claim that the usage of large format displays in a multi-user collaborative environment can increase productivity and the quality of work, but also to contribute to the field of project planning. Furthermore, the objective is to evaluate and understand how large displays usage influences people's cooperation and collaboration while performing a complex collaborative task. We study how the subjects actually interact and how different is their interaction, both with tools and other subjects, compared to the traditional approach.

We study the general problem of using large format displays in a multi-user collaborative environment for the general problem of project planning, with the specific problem of a university course planning. The traditional approach is represented by a regular whiteboard that will be compared with a project planning software developed for large format displays.

5.2 Tools and Material

In this section I will list and provide details about all the tools and materials that were use for the user study.

The location where the study took place is the Continuum room in the Electronic Visualization Laboratory at University of Illinois at Chicago. The room is equipped with a large format display running SAGE2, used for the technology approach, and whiteboards used to simulate the traditional approach. There are tracking cameras that were be used to track people movements to get insights on their type of collaboration.

Screen and Whiteboard

The display is composed by 6x3 tiled displays covering an entire wall of the room. Each display is 1.2m x 0.7m with 16:9 aspect ratio, for a total of 6,7m of length and 2m of height and a 32:9 aspect ratio. Each display has Full HD resolution, 1920x1080 making the whole wall display resolution of 11520 x 3240 pixels.

The whiteboard is composed by 3 whiteboards places horizontally each one 2.4m long for a total of over 7m of length and 1m of height. They provide enough space to perform such a scheduling task without limitations.

Software

Software used for task execution includes SAGE2 and the course planning application. SAGE2 allows users to use laptops to control and interact to the screen. Users need to use the SAGE2 pointer to interact with the application. The latter provide users with a grid with years and terms, and courses categories. The same grid was drawn manually in the whiteboard to create the same working environment by normalizing it as much as possible. Buttons allow the user to perform action such as adding courses, removing them and saving the session, as well as modifying courses. Text must be inserted manually for each course as the traditional approach counterpart will need to do this with sticky notes.

Other software has been used to asssit with the data analysis. First, the OptiTrack Motive application: it has been used to set and calibrate the IR cameras. Second, Unity3D: through a detailed 3D reconstruction of the room and the integration with OptiTrack data stream it was possible to have a digital reconstruction of the user study, and gather better insights on people interaction. Lastly, Matlab has been used to analyze data coming from the Unity logs and manual video coding to provide graphs and charts.

Finally, a tool to perform screen recording has been used to capture the recording of the Unity scene in 4K resolution, to use alongside with the raw data.

IR Cameras

Cameras with infra-red technology have been used to perform the tracking of people and objects inside the room. The cameras are OptiTrack Prime 13W and they provide 82 degrees of horizontal field of view and a resolution of 1.3MP at a maximum framerate of 240FPS. The room is provided with 24 cameras of this model and they have been specifically placed to cover the areas involved in the study, specifically two areas have been properly covered, one for the whiteboard approach and one for the large display approach. A pilot study was conducted to understand what portions of the space people were using and needed to be covered.

Some rigid body trackers have been placed on hats to perform the head tracking of the users. Trackers are composed of several (at least 3) IR reflecting markers placed on a rigid body. An object can be recognized when the markers are in the field of view of at least 3 cameras. They perform on-camera image processing to better recognize markers and thus, objects.

Data about the tracked objects then, is sent to Unity, so it can be associated to models inside the Unity scene.

Other equipment

Other equipment that has been used include hats for head tracking. On top of the hats markers have been placed with a personalized structure on 4 axes for better tracking.

Also, for the whiteboard version of the study, users were provided with all the necessary material to perform the task: sticky notes of 4 different colors, pens and whiteboard markers of different colors.

Finally, for video and audio recording a single wide angle camera has been used to record video in 1080p at 30FPS and audio. The camera was placed on a tripod for better angles and view.

Additionally, every participant was provided with a sheet of paper containing the description of the problem and all the information needed to perform the task.

5.3 The Complex Task

Participants were assigned a task to solve collaborating. In this section I will provide a description of such task.

The proposed problem is planning university studies of a bachelor of science (BS) in computer science and design (CS+DES) at UIC, scheduling courses from first year to the fourth year, for a total of 8 terms.

Courses are taken from real courses offered at UIC but for a program that does not exist yet, so the users are not familiar with all of the details. Users were provided with a set of courses and their task was to create a legitimate schedule. The list consists of a total of 37 courses to schedule, and it had all the necessary details on the courses. Courses were divided into 3 different categories. Each course has a term (either fall or spring) in which the course takes place, and days of the week and hours of the day the course is offered. A course could be offered in multiple terms and/or multiple days of the week, each one with its specific hour times. Each course might have some pre-requisites, co-requisites, or exclusive courses.

name	term	days	hours	prerequisites	corequisites	exclusive	credits	Category
Course A	Fall	MWF | TR	03:00 PM - 03:50 PM | 12:30 PM - 01:45 PM	Course B	Course C	Course D	3	Computer Science
	Spring	TR	09:30 AM - 10:45 AM	Course B	Course C	Course D	3	Computer Science

Figure 21: Example of a listed course to schedule

In Figure 21, a sample of how the course list is provided. Course A is held in both Fall and Spring terms. In the Fall term on two different sets of days, either Mondays, Wednesdays, and Fridays from 3:00pm to 3:50pm or Tuesdays and Thursdays at 9:30am to 10:45am. Following pre-requisites and co-requisites, Course B must have been taken before course A, course C before or while taking Course A, and course D cannot be taken if Course A is taken and vice versa.

There were some courses marked in bold, and those had to be chosen from a specific group of courses, there was a table for each group showing all the courses to choose from. All of the courses are 1 term long except for the ones belonging to one specific group that are two terms long.

The constraints that participants had to respect are:

- Maximum enrollment in 18 credit hours per term
- Minimum enrollment in 12 credit hours per term
- Minimum total of credit hours of 122
- Respect all prerequisites and co-requisites
- Total of 8 terms, excluded summers terms
- No transfer credits
- No overlapping classes

5.4 The innovative approach

We have seen that this study leads to the collection of qualitative data, requiring a qualitative data analysis. Normally, in such studies the generation of the codes is performed manually, with the help of dedicated softwares such as Atlas.ti [1]. This generally is a time consuming task. Depending on the quantity of codes to attribute to the data, to analyze one minute of data it can take over 10 minutes of manual review.

My approach tries to make this kind of analysis automatic and computerized, cutting down drastically the analysis time. Thanks to the tracking system combined with the 3D recon-

[1] https://atlasti.com/product/what-is-atlas-ti/

struction of the environment, all the codes related to spatial variables can be associated to the numerical data coming from the tracking system.

The tracked objects are: the two heads of the participants and the two tables used by participants. In this way they could freely move the tables around the room while still understanding if they were looking down at the tables.

Tracked objects have both x,y,z position and x,y,z rotation. Combining this information with objects in the Unity scene it was possible to determine what the subjects were looking at, their distances to the wall, to the whiteboard, to the other participant, and if they were sitting or standing. Different experiments made it possible to understand the parameters and tweak them to allow me to differentiate between these categories. Given the precision of the tracking system I was also able to get insights on the details of their visual attention and more detailed data on their distances, as explained in more detail in Section 5.8.

5.4.1 The unity program

A Unity scene has been created to recreate a virtual environment of the room where the user study took place. It's reconstructed to the finest detail to look exactly the same as the original room.

All the cameras of the tracking system has been placed inside the scene and, thanks to a script, it's possible to send the data coming from the tracked objects to the Unity scene. 3D models were to be manually adjusted to the tracked objects to represent their real world position inside the virtual world one.

From the data about the position of the heads it has been possible to differentiate the codes about distance to wall, in the technology approach, or the distance to the whiteboard, in the traditional approach, distance to the other participant, and whether they are sitting or standing.

Understanding where the user's visual attention was focused is not immediate and required a more sophisticated process. Targets of interest have been assigned with specific tags and colliders (Unity components of game objects) and I used the RayCast to retrieve what was the pointed object by each user's gaze in the scene. The combination of this information with the head rotation and position allowed me to create a precise mechanism to differentiate visual attention. Giving a different tag to each different display of the whole wall display allowed me to get an insight on the areas of the screen that were subject to the most attention. A similar technique was made for the individual whiteboards, where I can differentiate between the three different whiteboards composing the large one. It was also possible to determine what kind of document participants were paying attention to, whether papers or laptop.

In Figure 22 and Figure 23 it's possible to see the real room compared to the recreated one.

View

The main view is composed of 4 cameras: three orthographic view and one perspective view. The latter to catch a global view, that is similar to the real camera view and the first three to catch the precise positions, one top view, one side view and one front view. Each scenario, both display and whiteboard have their own dedicated cameras to better catch users movements.

Figure 22: Room of the experiments: real picture

Figure 23: Room of the experiments: recreated 3D model in Unity

A canvas that appears in the main view was built, displaying all the codes generated in real time for each participant. This allowed me to better tweak the parameters and catch coding errors when they happened.

I created two main scripts to allow the coding. First the "categorizer", that performs the coding and writes the result of each frame on the canvas, and second the "data logger" that logs all the data to a csv file for subsequent analysis with Matlab. The latter can be tweaked to log at different rates. In my experiments I collected 10 logs per second.

Each scenario has a slightly different version of both the scripts. Indeed I am not interested in all the parameters in both approaches but just some, for instance I don't need to collect the distance to the display while performing the whiteboard version and vice versa.

5.5 Hypothesis and Significance

The course planning application will allow people to create an admissible study plan, respecting all the constraints and not generating conflicts between classes times, and reducing the amount of time to find a solution. Thus, leading to reduced stress and complaints among the participants. We want to demonstrate that study planning, and thus project planning, is better using the help of a large format display and an adequate piece of software. Large format displays enable the resolution of common problems in the traditional approach, such as: the initial set-up and sketch, stick-notes legibility and dimensions, loss of stickiness. Indeed information are more clear, bright, and sharp. This leads to a better understanding of the problem through a more clear global view. Although, the software is less flexible than human creativity, such as doodling or displaying informations not planned and embedded in the software. My goal is to collect and provide useful information for future work and for the design of applications in the same field and similar environments.

We are interested in understanding:

- If this technology makes collaborative project planning easier compared to the traditional approach;
- More generally, if this technology makes multi-user collaborative tasks easier compared to the traditional approach;

- If the proposed application allows participants to have a better understanding of the information of the problem;
- How people interact in the two approaches, in terms of time spent interacting with other people and tools provided, and proximity both to other people and tools;
- What differences there are in the interaction in the two approaches;
- If leadership roles are taken by participants and how such roles differ in the two approaches;
- If the proposed application is missing critical features or is counterproductive compared to the traditional approach.

I expect people would prefer the software version rather than the traditional approach due to the clearness of the information displayed and a generally better view of the problem, as well the ease of performing some operations. Indeed, many of the manual operations are turned into simple mouse clicks in the software version. This should allow people to perform a slightly better planning in terms of time performance. Even though the difference in performance and the number experiments will not lead to statistically significant result. About the leadership role, I expect it to me stronger and more frequent in the traditional approach, and observing a more flat collaborative behavior, or having more role changing in the software version. This due to the different nature of the approaches: in the software the collaborators do not need to be in close proximity to the display as you need with a whiteboard, since you can perform all the actions from a desk with a laptop.

This experiment will help us confirm the effectiveness of technology, in our case large vertical displays, in a multi-user collaborative environment. More specifically, it will help us understand the advantages of such technology in the field of project planning, and how such technology affects people behaviors while performing a collaborative task. As mentioned before, the information obtained by this study could be used in future work on the design of applications in such environments.

5.6 Methodology

Every study run is composed of a group of 2 people and they perform the task in one of the approaches, either with the wall display or the whiteboard. In both approaches subjects are provided with desks facing the tool (whether screen or whiteboard) positioned about 3 meters distant, with the possibility to move it at the user's preference, and with the sheet of paper with rules and information about courses. Each group is provided with chairs for both participants. As mentioned in previous sections, the tables are tracked to better follow users movements. Since some aspects of the problem might be not completely understood, a facilitator represented by the PI is present in the room during the study and is able to answer questions that might come up.

5.6.1 Participants

Video recording, audio recording, and head orientation and position is performed during the session to capture the subjects' interaction with tools and other subjects. In order to perform the tracking participants wear a special hat with tracking markers. The entire duration of a session is approximately 60-90 minutes, including initial explanation and training. 6 such sessions per each condition (traditional or technology approach) have been conducted involving a total of 24 subjects.

5.6.2 Procedure

1. The PI will explain the purpose of the study to the subjects, describe the procedures, inform them of their rights, answer any questions that the subjects might have, and ask for their informed consent. Depending on the approach the procedure will be slightly different.

   Traditional Approach

2. The PI will explain how to use the materials that will be given to them. The material consists of:
   - A sheet of paper containing the list of courses of the study plan, the constraints they have to meet, and all the necessary informations about them.
   - A series of Post-its to attach to the whiteboard and markers, if performing the traditional approach.
   - A hat that will allow head tracking.
3. The PI will explain how a project planning session is done using the whiteboard.
4. The subjects will be given a 15 minutes training using the whiteboard and real sticky notes.
5. The material is given to the participants and the session using the whiteboard starts.

6. At the end of the study, the participants are asked to fill out a questionnaire related to the tasks they performed.

Technology Approach

2. The PI will explain how to use the materials that will be given to them. The material consists of:
   - A sheet of paper containing the list of courses of the study plan, the constraints they have to meet, and all the necessary informations about them.
   - A laptop connected to the internet that will be used to connect to the screen.
   - A hat that will allow head tracking.
3. The PI will explain how a project planning session is done using the project planning application.
4. The subjects will be given a 15 minutes training using SAGE2 and the the project planning application.
5. The material is given to the participants and the session using the wall display starts.
6. At the end of the study, the participants are asked to fill out a questionnaire related to the tasks they performed. This information will be used also to get suggestions on how to improve the project planning application.

5.7 The Collected Data

Collected data comes from five different sources.

Unity log

The "data logger" script of the Unity program is in charge of collecting a log about the calculated codings regarding position and orientation. The log is collected for all the participants and for all the possible different codings. In detail the log contains the following attributes:

- Participant: The participant ID marked with "Head A" and "Head B";
- Time: the time in seconds (with ms precision) from the beginning of the experiment;
- Date: the date in a custom format with ms precision;
- Visual attention code: the code for the visual attention computed in that instant; Possibilities are "wall", "whiteboard", "elsewhere", "other person", "document";
- Visual attention detail: this space provides detail about the visual attention, for display it provides the specific display id (ex. "Display 7"), for the whiteboard which whiteboard is being looked (ex. "central"), and for document the prediction of what type of document is being looked at;
- Distance to Wall Code: the code computed to categorize the distance to the wall, possibilities are "close", "medium", "far", "very far";
- Distance to Wall: numeric distance to the wall in meters;
- Distance to Whiteboard Code: the code computed to categorize the distance to the whiteboard, possibilities are "close", "medium", "far", "very far";
- Distance to Whiteboard: numeric distance to the whiteboard in meters;

- Sitting Code: the code computed to categorize sitting possibilities are ”sitting”, ”standing”;
- Height: numeric value of the head’s height in meters used to determine sitting code;
- Group Code: the code computed to categorize the distance to the other participant, possibilities are ”close”, ”medium”, ”far”, ”very far”;
- Group Distance: numeric distance to the other participant in meters;

Video Recording of the Coding

As described in Section 5.4 the Unity Scene has a canvas showing in real time what are the codings that the script attributes to the participants. This source is a secondary one; it is used only to verify the behavior of the ”data logger” whenever it fails to code correctly.

The unity program is exported as an executable and it is run on an external machine. A screen capture tool is used to record the Unity program in action. The resolution of the program is 4K as well as the recorded one.

Video Recording

An actual video recording of the real scene is performed. This is again a secondary source, that is used only as check when the previous application fails in coding. No manual coding is being performed directly on the video as the usual video coding approach for qualitative data analysis require.

The camera has a wide angle view and it’s positioned in order to capture the whole scene. Resolution is 1080p at 30FPS.

Audio Recording

The same camera is in charge of recording the audio source. In this case no automated tool was created to provide codings regarding audio, thus coding has been performed manually in this case.

Questionnaires

A questionnaire is given to the participants at the ends of the experiment. This helps to understand the users' thought and comments on the experience.

Two slightly different versions were created to match the two approaches. The questions have been kept as similar as possible to better compare the answers. In Table I all the questions are reported.

5.8 Data Coding

In this section I will cover the choices of codes for the data. As explained in Section 3.5 literature is divided mainly in two different approaches: the first, combines into a single unique coding visual attention and verbal communication, and the second instead performs different codings. The first approach generally presents a larger set of codes while the second has fewer but divided into different categories.

I decided to adopt the second approach and create different code categories with fewer codes each. The motivations behind this choice are the easier extraction of such codes, based more on a physical base and it requires less inference and abstraction. I took inspiration for the codes from the cited studies with some appropriate modifications to better accommodate the

TABLE I: QUESTIONNAIRE STRUCTURE

	Wall	Whiteboard
Q1	Please rate the ease of use of the study course planning application / whiteboard on a scale of 1 to 10, with 1 being difficult and 10 being easy	
Q2	Please rate the study course planning application / whiteboard interface with respect to the visualized information, such as their clearness and completeness, on a scale of 1 to 10, with 1 being not complete and/or not clear and 10 being complete and/or clear.	
Q3	Did you felt any lack or redundancy on the proposed interface with respect to the visualized information? If yes, list them in the space below	
Q4	Please rate the study course planning application whiteboard interface with respect to the offered functionalities on a scale of 1 to 10, with 1 being difficult and 10 being easy	
Q5	Did you felt any lack or redundancy on the proposed interface with respect to the offered functionalities? If yes, list them in the space below	
Q6	What pros do you find in the study course planning application with respect to the traditional way of planning a study course?	What pros do you find in the whiteboard approach with respect to a software version of planning a study course?
Q7	In general, do you think that a technology, such as a large display with a study course planing software, can help to tackle manual problems? (yes/no)	
Q8	If you answered "No" to question number 7, can you explain why technology can not help to tackle manual problems?	
Q9	If you answered "Yes" to question number 7, can you explain how the technology can help to tackle manual problems?	
Q10	Please, rate the ease of the whole process to find a solution with this approach, with 1 being difficult and 10 being easy	
Q11	Please describe any difficulties you faced while using the study course planning application / whiteboard	
Q12	Do you think this application could substitute the traditional approach with whiteboard and stickers? (Yes/No)	
Q13	Please write in the space below any further comments about the curse planning application used during this study	Please write in the space below any further comments about the Whiteboard approach used during this study

different study situation. I created 5 categories of codes that provide mutually exclusive and exhaustive states.

Visual attention

This code has been created to capture the abstract concept of visual attention of the participants. This code is not intended as the actual visual focus of the subjects, as it's a more physical parameter, defined as the direction in which the head is pointing.

- Display: the participant is looking at the display;
- Documents: the participant is looking at the documents, either papers or laptop;
- People: the participant is looking at the other participant;
- Elsewhere: the participant is not looking at something related to the task or has a passive attitude to the problem;

Display or Whiteboard Distance

Distances have been divided in 4 codes. Close is the distance within which it is possible to physically interact with the tool, for instance when writing on the whiteboard or moving sticky notes, it's approximately the length of an arm. Medium is a range of distances that provides a detailed view while it's too close to get a global view, this is the distance within it's possible to clearly view the information content of sticky noted or small text on the display. Far provides a global view, at this distance small details are lost. Very far it's when loss of detail occurs.

I expect the wall display to be exploited best at a "Far" distance due to clearer information, while the whiteboard at "medium", since "Far" could be too distant to catch all the details.

- Close: range between 0m - 0.7m of distance;
- Medium: range between 0.7m - 2m of distance;
- Far: range between 2m - 4m of distance;
- Very far: over 4m of distance;

Sitting

This code categorize the sitting or standing behavior of subjects.

- Sitting: the participant is sitting;
- Standing: the participant is standing;

Subject Distance

This codes adopts the same distances used for display and whiteboard. A quiet conversation happen at a close distance, within it's also possible to share information, for example looking at the same sheet of paper. It's also the distance that includes the "intimate zone", indeed I expect friend subjects to spend more time at this distance than unknown subjects. Medium is the distance within a regular conversation takes place and Far it's not an optimal distance for information exchange except particular cases, for instance a subject dictating form sit place at the desk times and days of a course to another subject that is writing them into the whiteboard.

- Close: range between 0m - 0.7m of distance;
- Medium: range between 0.7m - 2m of distance;
- Far: range between 2m - 4m of distance;
- Very far: over 4m of distance;

Verbal Communication

Verbal communication codes have been created to understand verbal communication behavior among participants. In other studies no differentiation between who is driving the conversation is performed, for different reasons including the higher number of participants. In this study due to the low number of participants, I decided to differentiate between who is speaking to better understand and detect unbalances in verbal communication. This will also help getting a better picture when someone takes a leadership role.

- Silence: no one is talking;
- A Talking: A is driving the conversation. For instance, explaining, dictating or sharing an idea;
- B Talking: B is driving the conversation. For instance, explaining, dictating or sharing an idea;
- All Talking: quick switch of talker. The conversation is not evidently driven by either A or B, but it's a flat, quick switching, exchange of ideas or discussion;

5.9 Results

In this section I will report all the data collected during the experiments and discuss it.

Twelve different sessions have been conducted, six per condition. Study numbers go form 1 to 12 and for convenience all odd studies are performed in the wall display approach, and all even numbered studies are performed in the traditional approach with the whiteboard. Each study involve two subjects, that I will address as "A" and "B".

5.9.1 Scheduling

Regarding scheduling quality there is no formal distinction between legal schedules. No rate or points were assigned to differentiate between schedules, thus, all schedules that respected the requirements and constraints were considered equally valid ones.

Time was the only performance measurement attributed to the studies and they are reported in Figure 24. Average times are also reported in Figure 24 with an average of 57.33 min for Wall approach and standard deviation of 6.25 min, and an average of 63.00 min for the Whiteboard approach and standard deviation of 14.23 min.

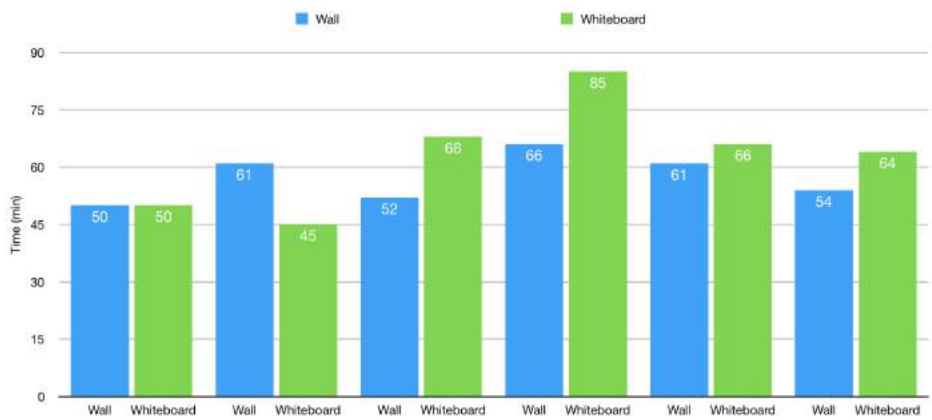

Figure 24: Duration of individual studies.

We can say the wall approach has performed slightly better and with less sparse results but we do not have a statistical significance to claim it made performing the task faster. The small

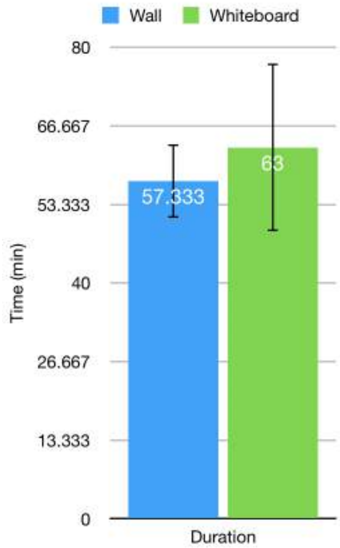

Figure 25: Average of duration of the studies.

time difference could be attributed to some manual tasks that in the Wall version have been substituted with automated tasks, such as the duty to physically go to whiteboard to add a course or move one. In addition, typing on a keyboard is generally faster than writing with a pen.

A reason why timings are similar is due to the task itself. The given task, indeed, is a highly complex one, that requires a high cognitive load and time to process all the information provided. Thus, the advantages of the technology might be mitigated by the complexity of such a task.

The fact that results for the Wall version are comparable to the traditional approach in terms on timings are an indicator that the wall software can be a substitute of a whiteboard and thus, a valid technology for completion of planning tasks.

5.9.2 Visual Attention

Firstly, before analyzing the data it's worth mentioning that generally, in this kind of coding it's common to have a general code that ensembles all the other codes when they are quickly changing in rapid succession. This sort of mixed code is reasonable to have when dealing with manual coding, especially for long tests analysis. In my approach I did not include such a code since the system I employed allows me to differentiate when quick a successions of visual attention occurs.

In Figure 26 all visual attention times codes are reported.

As it is noticeable form this chart, some studies have similar configurations, but others present highly different behaviors from the subjects. In Section 5.9.9 I try to group them into clusters of similar study runs and point out differences between these clusters to catch differences or similarities in collaboration and what are the causes of such.

For completeness in Figure 27 the average times are reported in a comparison between the two approaches.

There are some minor differences between the two: first, document attention is noticeable, subjects spent about the same time on the visual attention to documents, 44.3% and 47.4% respectively for wall and whiteboard. This is a reasonable result since the amount of information to analyze is the same and requires the same amount of effort in the two versions. But it took

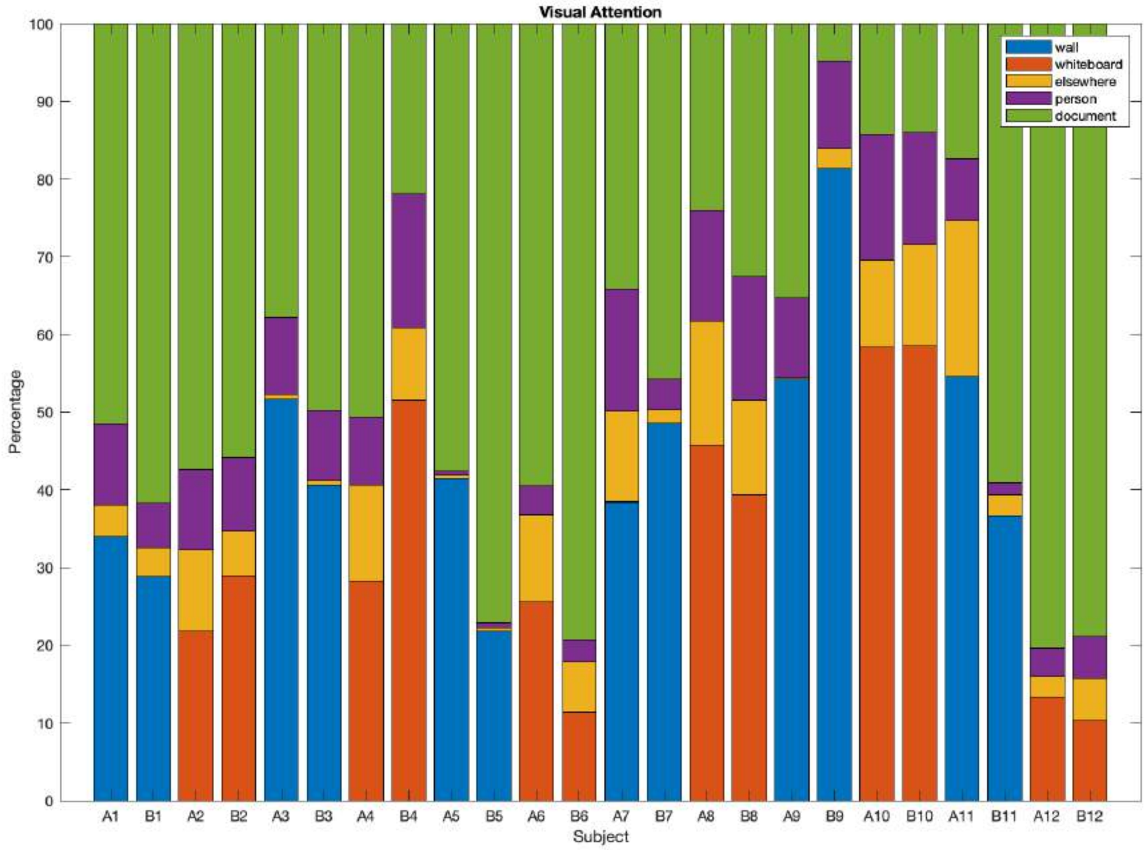

Figure 26: Visual attention for each participant for each user study. Participants are marked with a letter ("A" and "B"), studies are numbered with numbers. Odd studies are wall approach and even studies are whiteboard approach.

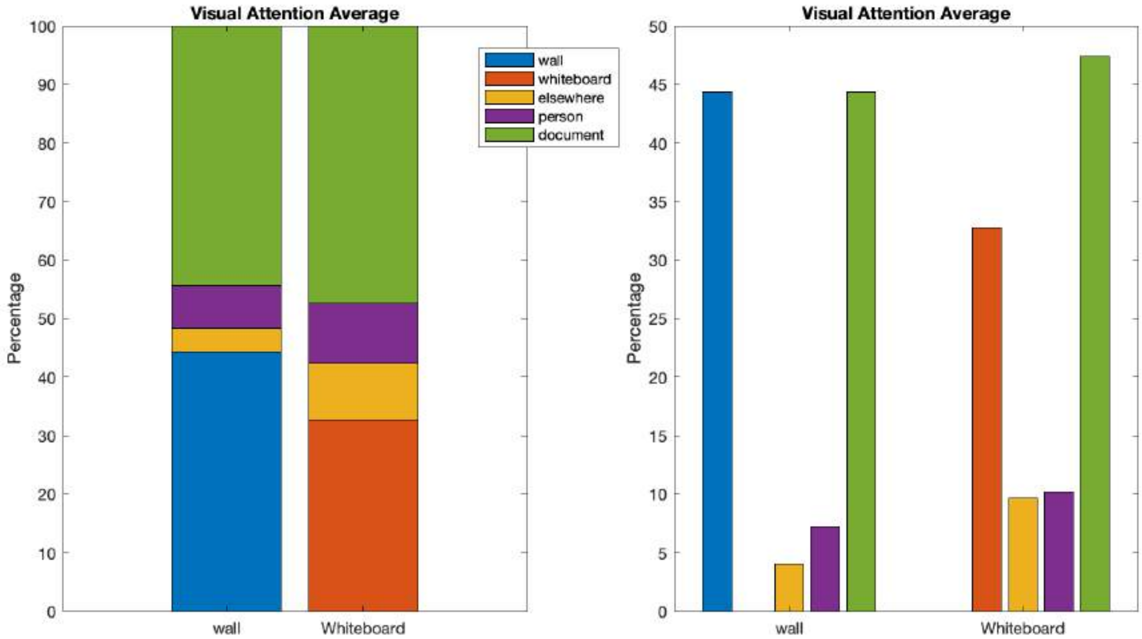

Figure 27: Visual attention average comparison between wall and whiteboard approach.

slightly different times for attention at the offered tool, 44.4% and 32.8% respectively for wall and whiteboard. So, in the wall approach people spent, on average, more time looking at the wall than they did in with the whiteboard. This is a positive indicator of the fact that clearness and legibility are better in the wall version, leading to a clearer overall representation of the problem. Also, people spent more time with visual attention on each other in the whiteboard approach than in the wall approach, with 7.6% with wall and 10.2% with whiteboard. Also confirming the previous statement, due to the better clarity of the display people had to discuss less on the problem. A similar trend occurred for the "elsewhere" code, with 4.0% for the wall and 9.7% for the whiteboard. This can be justified by the fact that people in the wall approach preferred to stay sit almost all the time, while with the whiteboard people had to stand and move and leading their visual attention elsewhere while moving.

To better understand the differences and similarities in collaboration styles and dynamics, we can divide the experiments into clusters of similar data and compare the approaches between such clusters.

To better view the individual clusters in Figure 28 I plotted the complete data, sorting it by experiments based on time amount of each category. To generate the plot, the sum of the values of the two subjects per each experiment have been calculated and compared between all the experiments. In each graph just one category is taken into consideration, for instance, for document attention, or the graph at bottom left, I obtained 12 values, one per experiment (each value is obtained by the sum of document attention of subject A and document attention of B) and then I obtained the list of the sorted experiments. The chart is showing the same data as the initial graph in Figure 26 sorted according to such a list. The procedure has been repeated for all the other categories, wall/whiteboard attention, people, elsewhere generating 4 of such graphs. A similar procedure has been applied to Figure 29 with the only difference that instead of the sum I calculated the difference between the visual attention times of the two subjects. This helps to better visualize witch are the experiments with the highest differences per each category.

From the image it's possible to identify the following clusters: {5,6,12} present significantly long times for attention on documents, while on the contrary {8,9,10} have the shortest. All the others {1,2,3,4,7,11} have an average amount.

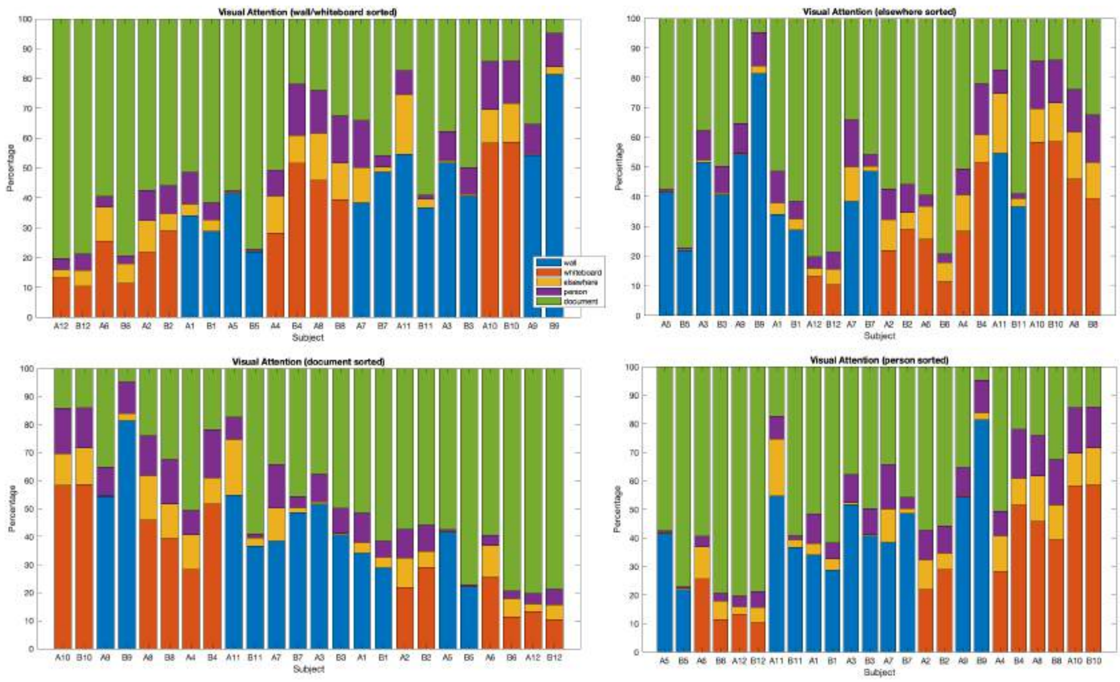

Figure 28: Visual attention comparison between sorted data in the 4 main categories.

About the document attention it's possible to form the following: {9,10} with the highest, followed by {3,4,7,8,11} with an medium amount, and {1,2,5,6,12} with a low amount.

Other possible clustering are reported in the following table Table II.

Furthermore, I sorted all the graph according to how different (or similar) the visual attention is between the two subjects of experiments in terms of all 4 categories. This will help find other clusters with a different meaning. In this case I will group together studies that have very different behaviors between subjects and studies that have a very similar behavior between the two, with the results shown in Figure 29.

TABLE II: CLUSTERING STUDIES BASED ON VISUAL ATTENTION.

	Short	Medium	Long
Document	8,9,10	1,2,3,4,7,11	5,6,12
Wall/wb	1,2,5,6,12	3,4,7,8,11	9,10
Person	5	6,11,12	1,2,3,4,7,8,9,10
Elsewhere	3,5,9	8,10,11	1,2,4,6,7,12

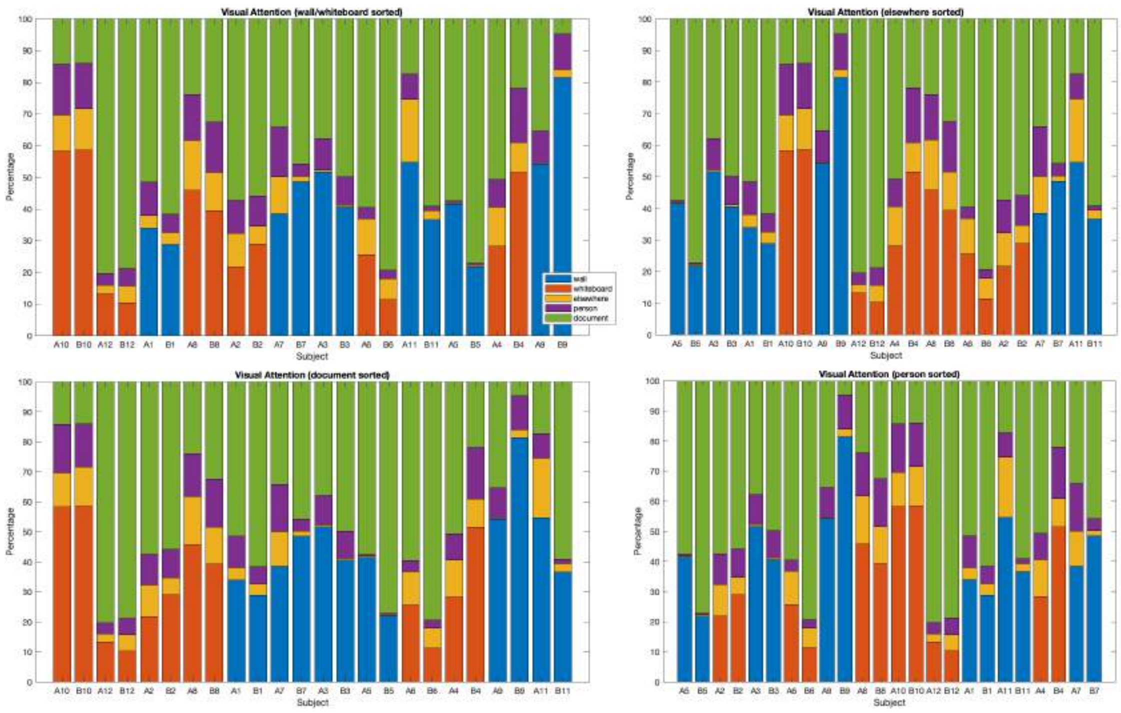

Figure 29: Visual attention comparison between sorted data based on the difference in subjects' visual attention in the 4 main categories.

Thus, it is possible to notice that the most balanced behavior between subjects is in experiment 10 and 12, being at first positions both in wall/whiteboard and document attention. The most different studies are 11,9,4 for document attention and 9,4 for wall/whiteboard. Study 11 appears also as the most different in the elsewhere category, and the third most different in person attention.

Other remarkable facts, experiments 10 and 9 appears in the same cluster multiple times in the first clustering, as the longest wall/whiteboard attention and the lowest document, appears in two very different clusters in the second clustering, indeed 10 is one with the most similarities between subjects while 9 one of the most differences.

Also, experiment 11 appears always in an average times in all categories, has the most unbalanced behavior of subjects. Similarly, but less evidently, 5 and 6 show the highest document time and the lowest wall/whiteboard time, while being categorized as some of the most different in subject behavior. In addition 5 is the absolute lowest in person attention and elsewhere attention.

TABLE III: CLUSTERING STUDIES BASED ON THE DIFFERENCE IN SUBJECTS' VISUAL ATTENTION.

	Similar	Average	Different
Document	2,10,12	1,3,7,8	4,5,6,9,11
Wall/wb	10	1,2,3,7,8,12	4,5,6,9,11
Person	2,3,5,6,9	8,10,12	1,4,7,11
Elsewhere	1,3,5,10	2,4,6,8,9,12	7,11

Sorting the experiments based on the overall difference of times in all categories we obtain the following ranking in Table IV and represented in scale in Figure 30:

TABLE IV: OVERALL DIFFERENCE IN SUBJECTS' VISUAL ATTENTION ON THE SAME EXPERIMENT.

Study	11	4	9	7	6	5	3	8	1	2	12	10
% diff	41.8	31.8	30.5	21.8	19.8	19.6	12.0	10.2	10.1	7.1	4.5	2.0

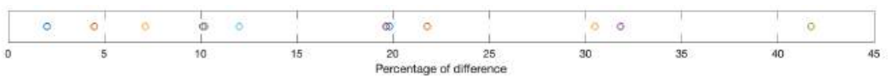

Figure 30: Absolute differences representation in scale.

This confirms that experiment 11 is the most different in terms of subject visual attention between the same experiment, following by 4, and 9 very different, and 2,12,and 10 the most similar.

A different kind of analysis can be conducted to spot analogies and differences in visual attention between the two approaches. The average visual attention times, the same as Figure 27, of both approaches have been divided into 8 different bar graphs. Each graph shows the average data of an octave of the time of the experiment, from the beginning to the end. The bar graph is shown in Figure 31.

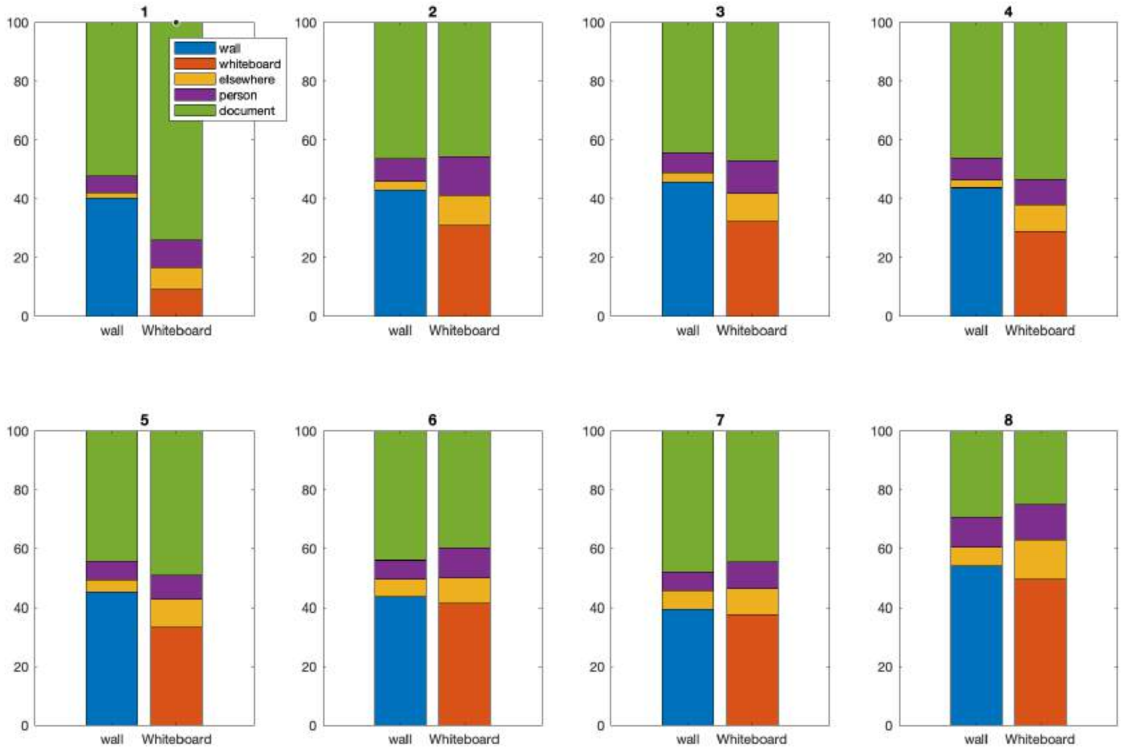

Figure 31: Comparison of the average visual attention of both approached for each category over time. Each graph is named after the octave they represent.

Another view of the problem is shown in Figure 32. The same principles have been applied but to a full plot of the data. The graph has been constructed in a similar way I created the above mentioned bar chart. But, instead of dividing the experiment into 8, I took 60 points, roughly one per minute, considering the average experiment to be about 60 minutes. The graph has been smoothed with a ”lowess” standard Matlab function for illustrational purposes. The smoothing doesn’t affect the meaning of the data. The graph has been created also to catch “bursts” in the data that may have not appeared in the 8 bar graph. But there were no “bursts” since all the trends were on average flat, thus, the graph just gives another, more detailed, view of the bar chart.

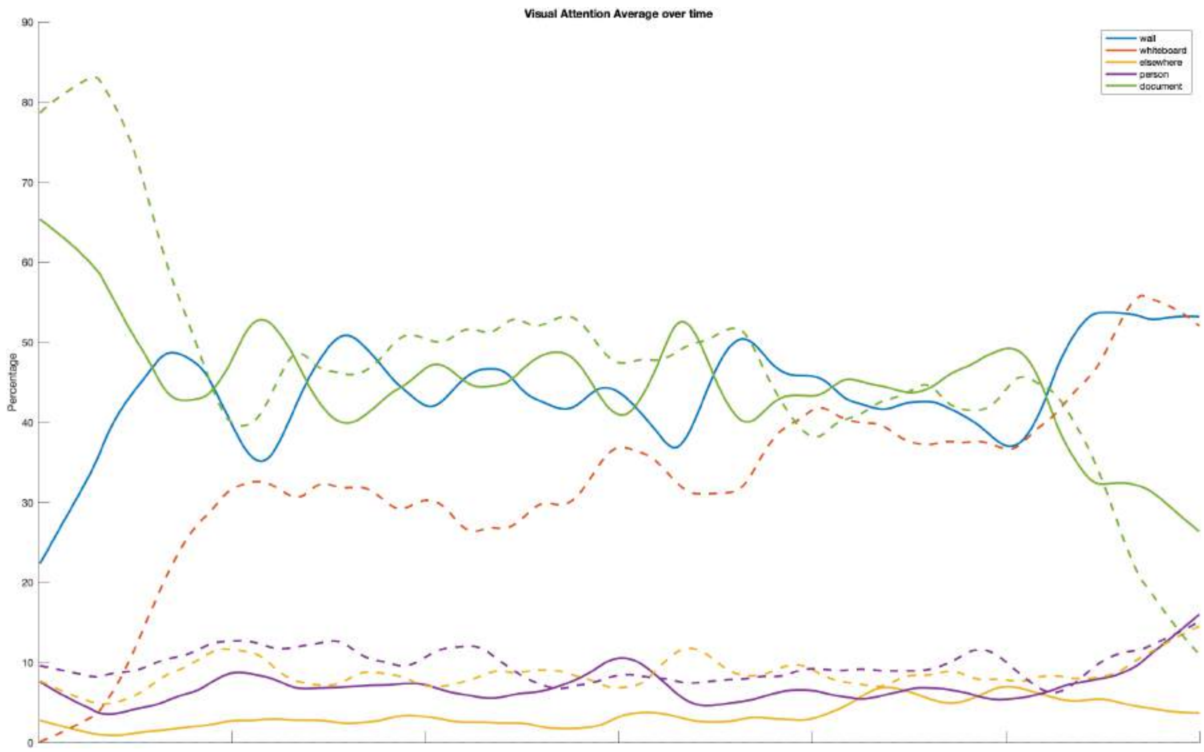

Figure 32: Comparison of the average visual attention of both approached for each category over time. Wall approach is characterized by a continuous line, and the whiteboard by an interrupted line.

This graph allows us to catch details taking into account the time factor. New differences appear: wall attention is high since the beginning of the experiment while whiteboard starts lower and gets higher in later phases. Document attention is also lower in the display approach, but just in the first moments, and reaches about the same level from the second octave. Other people's attention also is more pronounced in the whiteboard approach, especially in the first half. In light of this data I can state that the display affects visual attention in respect to the traditional approach.

All the data shown in this section is reported with detail in Table V for completeness.

Calculating the coefficient of correlation between study duration and visual attention times, no strong correlation was found. Coefficients are: 0.11 between duration and wall/display times, 0.33 showing a weak uphill linear relationship with "elsewhere" attention times, 0.3 with person attention times, and -0.23 with document time. Also, the coefficient between study duration and absolute difference of attention times between subjects on the same experiment previously reported is -0.32 showing a weak downhill linear correlation.

Analyzing further data will provide a clearer image.

5.9.3 Display and Whiteboard distance

Distances code meaning are discussed in Section 5.8. During the experiments subjects were told to have absolute freedom of movement, and also that they could change the position of the table. The table was also tracked by the tracking system and this allowed a correct prediction of visual attention even when the table position was changed.

TABLE V: DATA TIME OF VISUAL ATTENTION PER SUBJECT IN PERCENTAGE.

Subject	**Wall/wb**	**Elsewhere**	**Person**	**Document**
A1	34.0	4.0	10.5	51.5
B1	28.8	3.7	5.8	61.7
A2	21.8	10.5	10.3	57.4
B2	28.9	5.7	9.5	55.9
A3	51.6	0.7	9.8	37.8
B3	40.6	0.5	9.0	49.9
A4	28.3	12.4	8.7	50.6
B4	51.5	9.3	17.3	21.9
A5	41.5	0.3	0.6	57.6
B5	21.9	0.4	0.7	77.1
A6	25.6	11.2	3.7	59.5
B6	11.3	6.6	2.8	79.4
A7	38.4	11.7	15.7	34.2
B7	48.6	1.7	3.9	45.8
A8	45.8	15.9	14.3	24.0
B8	39.4	12.1	15.9	32.6
A9	54.2	0.2	10.3	35.3
B9	81.4	2.5	11.3	4.8
A10	58.3	11.3	16.2	14.3
B10	58.6	13.0	14.3	14.1
A11	54.6	20.0	8.0	17.4
B11	36.6	2.8	1.5	59.1
A12	13.3	2.7	3.7	80.4
B12	10.3	5.3	5.5	78.8

In the experiments with the wall, since there was no need to physically move, people preferred to remain seated during all the length of the experiment in a far position. This allowed them to exploit all the space offered by the display and thus, to have a global view of all the courses at once. In the whiteboard approach subjects were obviously obliged to stand and to walk to the whiteboard. For completeness I report the graphs of the wall distances in Figure 33 and for the whiteboard in Figure 34

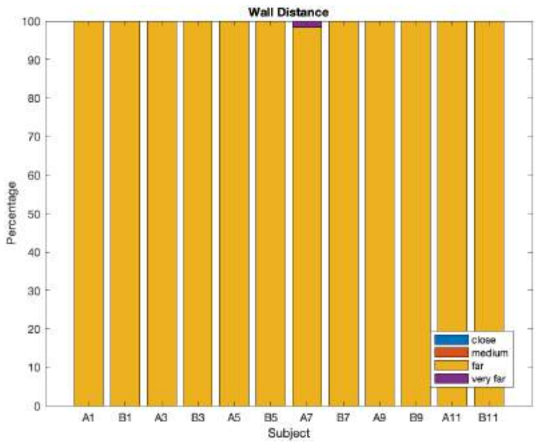

Figure 33: Distances of subjects from the Wall Display.

As we can see from the whiteboard distances chart, there are some differences in the approach to the problem. Some subjects spent a significantly larger amount of time at a far distance (experiment run number 6 and 12) than others with a high amount of time spent at a close distance, so in proximity of the whiteboard, (experiment run number 8 and 10). Further-

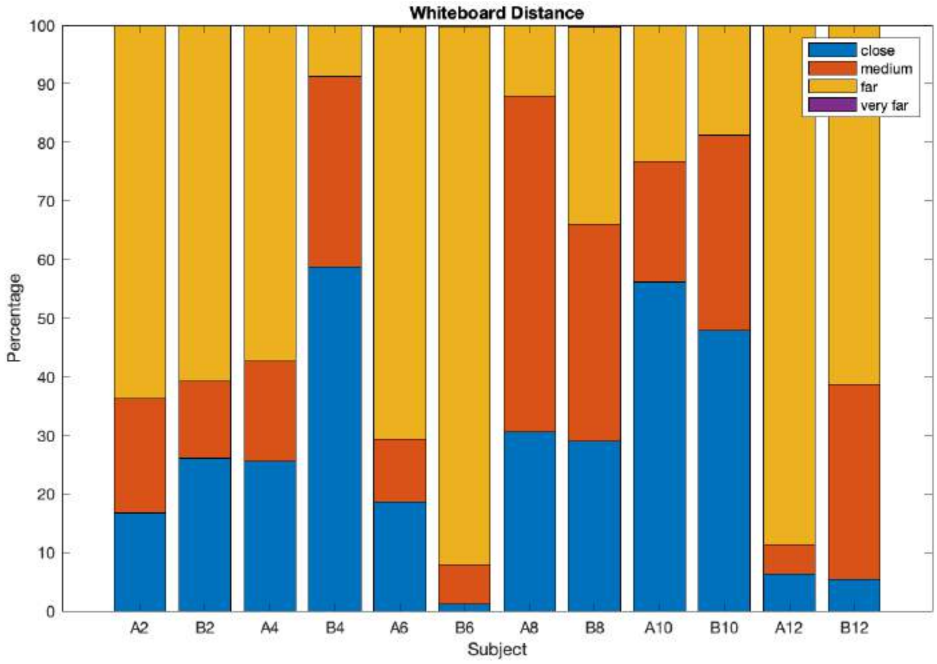

Figure 34: Distances of subjects from the whiteboard.

more, in experiment run number 6 and especially in number 4 there is a significant difference in the distances to the whiteboard between subjects. This behavior is explained by a differences in "leadership role", indeed in those experiments one subject stayed mostly at a distance while the other had the "control" of the whiteboard and was in charge of physically placing courses into the schedule. But this counterintuitively happened in two opposite scenarios: indeed in 4 the "leader" stayed in place and was dictating moves to the other participant that stayed at the whiteboard to execute the choices of the leader, while in 6 the leader was the one going to the board to apply changes, while the other participant stayed in place.

In experiment 12 there was no leadership dynamic and collaboration was balanced between subjects. They both stayed most of the time in place working on the documents, and when it was time to update the whiteboard they did it together. This explains why, compared to 6, their distances are more similar to each other.

With these charts it's already possible to state a difference in the working style of display and traditional approach: distance from the tool is a relevant parameter that can be influenced by the collaboration style adopted by participants, in the traditional approach, while it's not a relevant parameter and no insights on collaboration style can be deducted from it in the display approach.

Merging this data with the sitting times will give a wider view on this behaviors differences.

5.9.4 Sitting

In the chart in Figure 35 sitting and standing times are shown. As we can see in all the odd numbered experiments, i.e. the wall display ones, people remain seated all the time.

Looking at the whiteboard runs (the even numbered experiments) we can see that sitting and standing times vary on different experiments. Experiment number 2 showed a more balanced approach in terms of leadership role and whiteboard control exchange, in this run indeed there was no evident leader and collaboration style was mostly "flat", with ideas and active discussion coming form both participants. Looking at the distances to whiteboard, we can see indeed similar close and far distances times, confirming the balanced type of collaboration. Experiment run 4, on the contrary has an unbalanced time between subjects in both charts: subject A was sitting and at a far distance more time than subject B, who stayed at a close distance and standing for more time. In this case the collaboration type was unbalanced and leadership role took place, confirming the data of whiteboard distance.

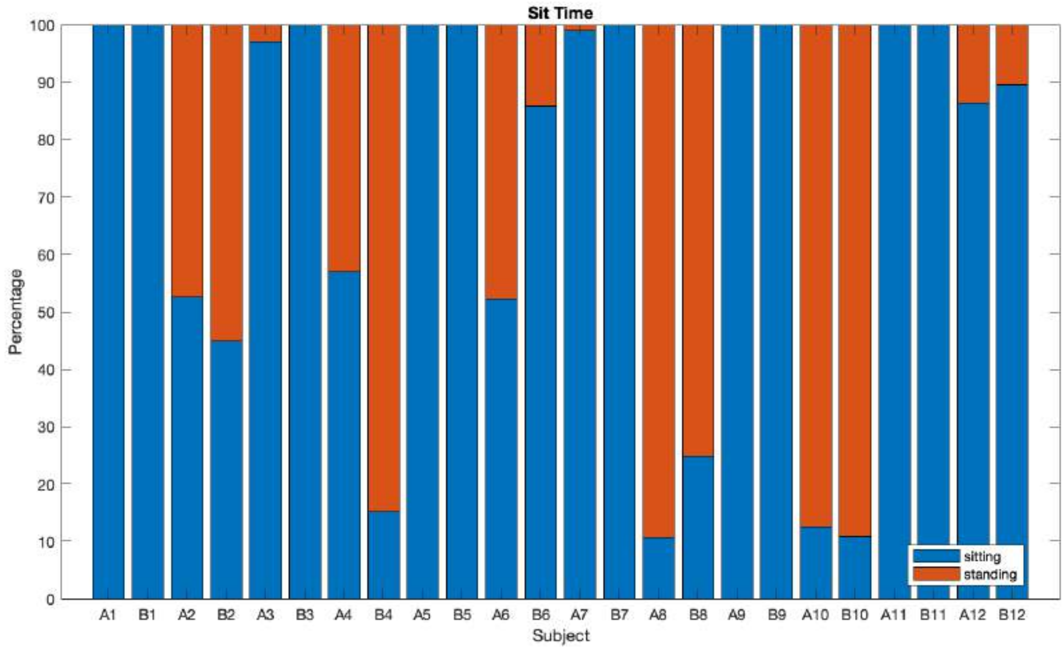

Figure 35: Sitting and standing times.

Similar differences in role taking and collaboration style took place also in the wall display approach, though it's not possible to find the confirmation from this data, but I will need to find it from the visual attention and verbal communication data, since physical movements in the wall experiments were limited.

5.9.5 Subject distance

Another physical parameter that can add additional information detail is the distance between subjects. In Figure 37 distances are reported showing very static results for the wall approach as expected.

Subjects in the wall approach (even numbered experiments) spent most of the time at close distance. The only "out of trend" results were given by experiment 5. But taking a look at their distanc graph in Figure 36 it's possible to see that they stayed around the edge between "close" and "medium" distance, making this result in line with all the others.

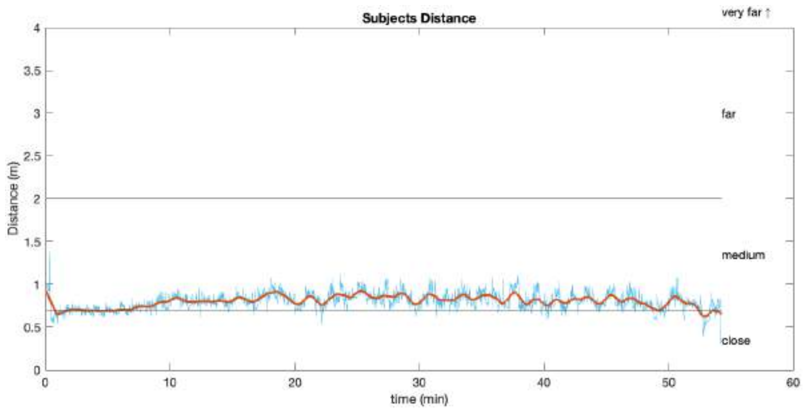

Figure 36: Distances between subjects over time for experiment 5.

Switching to the whiteboard, there is more variation in the data. Trying to cluster the data according to similarity, we can place study {4,6} together and {2,10,12} with the 8 being a middle way between the two clusters, but closer to the second one.

Cluster 4,6 goes alongside the previous results, confirming the leadership role dynamic that took place in such experiments.

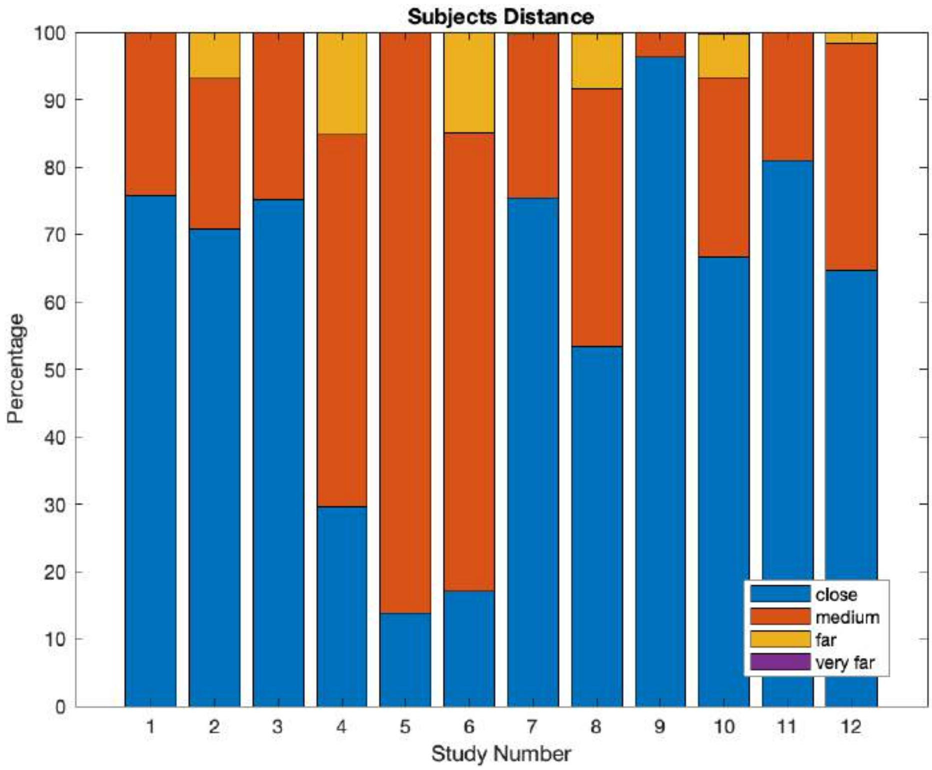

Figure 37: Distances between subjects per experiment.

A similar consideration as in Section 5.9.3 can be stated: subject distance is not a relevant parameter and no insights on collaboration style can be deducted from it in the display approach, while it is possible in the traditional approach.

5.9.6 Verbal communication

Verbal communication is the last relevant factor to analyze differences and similarities in collaboration. This is the only code that has been coded manually, as the standard approach. To better see the differences I divided the codes between the two participants. I expect more unbalanced verbal communication in experiments where leadership role dynamics took place. The bar chart of audio codes can be seen in Figure 38.

As can be noticed in the chart, experiment 5 has a significantly larger amount of time spent in shared conversation, and low of silence. This case has appeared to be an edge case also in the other data analyzed. This is a good example showing that verbal communication and visual attention to each other are not related parameters, indeed the participants spoke to each other most of the time even if they barely looked at each other.

Between all the other experiments some similarities and differences can be found and some clusters can be formed. Indeed the experiments that have the most differences between "A talking" and "B talking" are {1,4,6,7,11} and in most of them leadership dynamics took place. More details are discussed in Section 5.9.9.

Furthermore {3,12} have significant longer times of silence than the others, and {7,11} have a slight more time of "all", thus conversation, of the others.

In Figure 39 the averages of the audio coding are shown.

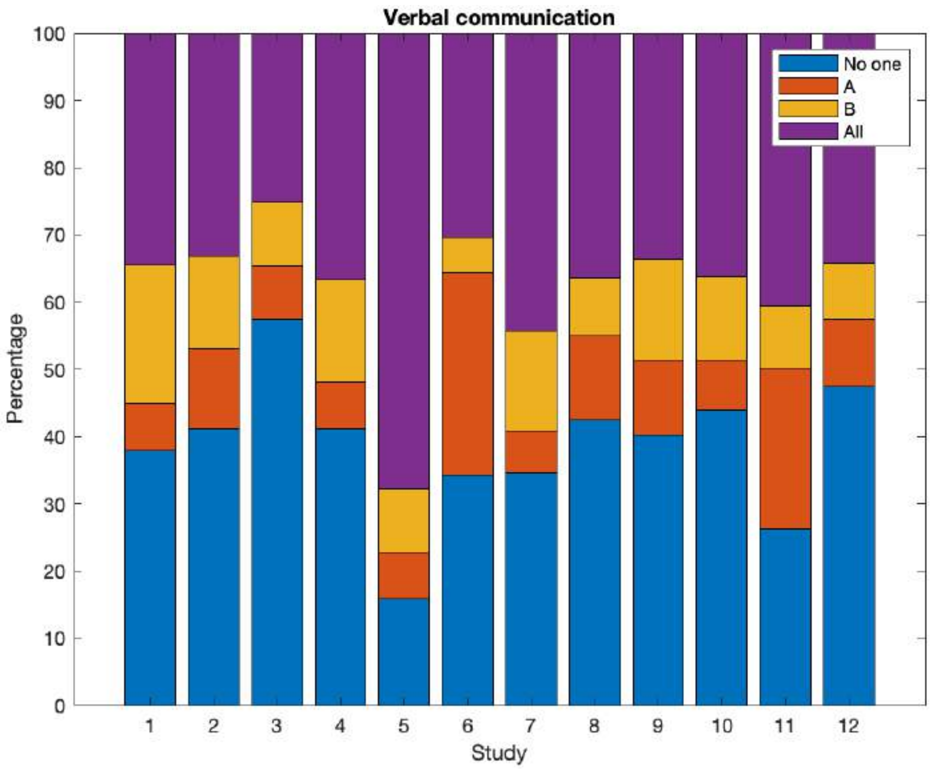

Figure 38: Audio coding of each experiment.

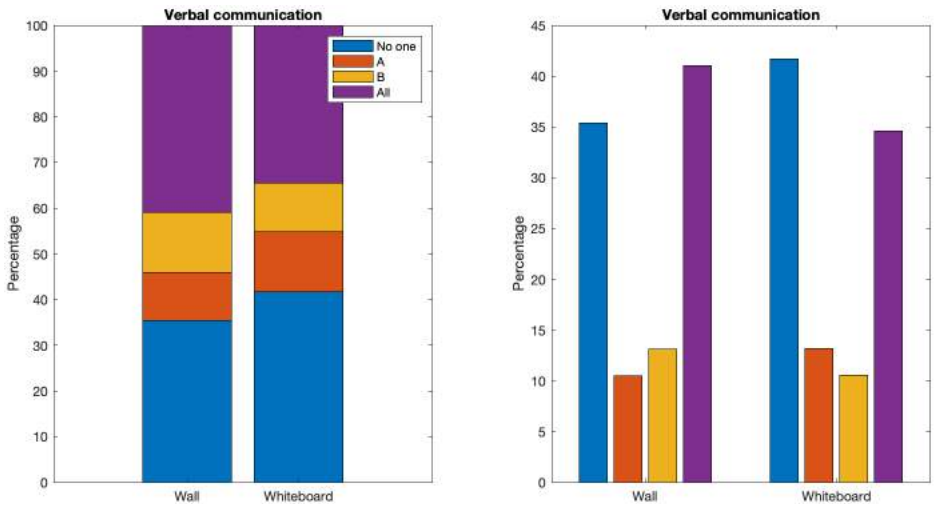

Figure 39: Average of audio coding.

Almost no difference is found for "A" and "B" talking between the Display and the Whiteboard approach. But, a slightly larger amount of time is spent in silence in the whiteboard approach and slightly more time having conversations on the Wall. This could suggest that the display approach helps and incentives the discussion and ideas sharing, even if no statistical evidence is found.

5.9.7 Questionnaire answers

Questionnaire answers are a instrument alongside the data that can help frame users thoughts and feelings about the experiment. Some bugs in the software have been found and some interesting aspects have been highlighted.

Answers to questions with a rating have been reported in Figure 40.

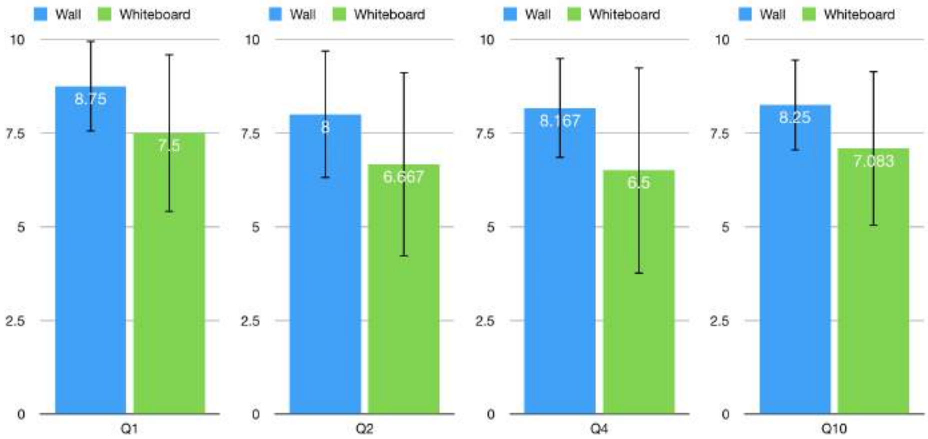

Figure 40: Mean and variances of answers to rating questions.

Even though results are certainly biased by the excitement for the usage of such technology, some interesting results appeared. Users reported finding the wall usage easier than the whiteboard, providing more complete information and better functionalities, with no statistical significance, though. But this means that the software combined with the technology was at least able to provide the same level of functionalities and overall experience, without missing any critical features.

Interestingly users that used the Wall version found the overall process to find a solution easier that in the whiteboard, again with no statistical significance.

To Q7 just 4 users found that such technology couldn't help to tackle the manual problem, 3 of which used the actual wall. These results contrast with the answer to Q12, where all users answered "Yes" claiming that the used application could substitute the traditional approach.

Based on all the answers received on the open questions, in the following I summarize some criticism that emerged: chosen color coding doesn't help the scheduling, the menu is too far from the pool lane, from the grid layout it was not intuitive enough to understand if a course was offered in Fall or Spring, there was no automatic count of credits, nor dependency visualization, a user also stated that the UI is not intuitive, and days could be highlighted more. Two users didn't understand at first that dragging courses also was "behind the scenes" changing the term or category, and finally the last criticism is that such technology is not available to everyone. A user stated that there are no advantages over a laptop. An interesting comment is that there was no time visualization beyond a term, the user is suggesting to add a more fine view of time, a way could be the one showed in Section 4.4.2.

Some bugs on the software were spotted: sometimes clicking on a task wasn't opening the modification pop up, constricting some users to delete and recreate some tasks. When deleting a task a double click on the Yes button was necessary, no cursor is present when writing text into the form, and sometimes the button response was slow. Also, just while dragging some courses it may happen that the dragged one was behind some others in the grid.

Some features were proposed: automatic credit count, help button, constraint visualization, an integration/combination with phone and tablet, and alert when courses overlap. Though the help button is a good idea, the others were improvement that would have resulted in a less fair comparison with the whiteboard.

Some criticism about the whiteboard: the need of creativity, small text and unclear visualization, editing and creating was too long, the need to stand to have a clear view, it can become messy very easily, and difficulty in visualizing all the constraints together, finding a correspondence between courses on the whiteboard and the provided list, not having a predetermined grid, and finally that one has to be present at the board.

Some advantages of the whiteboard: free placing, easy to use and to navigate.

Some advantages of the wall display application: ease of adding and modifying, clearness of view and legibility was a frequent positive comment, saving, clearness even from far, organization, some users found useful the color coding by category, time saving, dragging feature, ease of collaboration, sense of a global view (frequent comment), and finally it allows to focus better on the scheduling avoiding contour activities.

Overall people enjoyed using the application and found it provided at least the same functionalities as a real whiteboard, and generally a clearer and a more global view of the problem.

5.9.8 Different Strategies

Analyzing the video and audio records it's possible to notice that not all the groups adopted the same strategy. Indeed some different patterns came out, and I will list all of them trying to relate them to the previous analysis and groupings.

The most common strategy was addressing the problem "vertically". In this approach users tried to fill up completely a term before going to the successive terms. Making this a time-first strategy. This approach was adopted by experiment {1,2,3,5,6,11,12}.

A less common approach was addressing the problem "horizontally". In this case users instead of filling up a term they tried to place all the courses of a certain category, and then switched to the other categories one at a time. This is a category-first strategy. This approach was adopted by experiment {9}.

A mixed technique appeared in experiment number 4, combining both vertical and horizontal approaches together. In this case the horizontal approach stopped at half of the time, at the end of year 2. They filled up all the courses of the first category that could fit in the first two years, before switching category. When the first 2-years block was filled the same procedure was applied to the second 2-years block. This experiment was conducted on the whiteboard and it lead to the shortest time, beating also the fastest experiment on the wall. This mixed horizontal-vertical technique was also combined in another very effective technique when dealing with constraints, dividing the experiment in two phases. In the first phase all courses were placed into the whiteboard, using the mixed technique, and paying attention to dependencies in term of pre-requisites, co-requisites, and exclusive courses. When all the courses were in place, the second phase started. In this phase time checking was conduced leading to some movements when conflicts arose. This combination of techniques was revealed to be very effective and it lead to the best performance. Unfortunately, no wall display experiments were conducted using such techniques, but it would have been interesting to make a comparison.

Also in experiment number 7 a mixed technique was used, though different. In this case more freedom was adopted when switching categories.

Finally, in experiments number {8,10} no technique that could be matched with the previous ones was adopted.

5.9.9 Similar experiments and Edge Cases

Experiment number 12 had similar visual attention between subjects, similar sitting times and whiteboard distances, and balanced collaboration happened. This suggests that on the whiteboard approach there is a correlation between the similarity of codes between subjects and collaboration style of balanced.

Experiment 5: This is a particular case in which fluent collaboration took place and no leadership dynamics happened, even though they never shared sight of the same document, each one just looked at their own, and never looked at each other to talk. Indeed looking at the visual attention chart it's possible to notice an abnormal trend as well for experiment 5: subjects barely looked at each other and they spent most of their time looking down and thus, focusing their attention at the documents, and looking at the wall almost all the rest of the time. This will be an example, after the integration with the verbal communication data, of non-correlation between verbal communication and visual attention.

Experiment 10: this experiment is the only one where evident signs of frustration appeared, and it is the one with the longest visual attention on other person, elsewhere, and whiteboard. It's also the one that appeared to have the most similar visual attention between subjects, with an overall difference of 2%. It is the one with the most time spent on close distance to the whiteboard, and accordingly one of the highest standing times. This experiment has been clustered and associated in previous section to many other similar experiments, but when looking at the structure of their collaboration it appears very different from the others. No evident leadership role was taken, but it is the only experiment where division of work was observed. Subjects picked a couple of courses and they assigned one each to schedule independently. Their strategy was not identifiable inside one of the classes of Section 5.9.8, but the closest could be "vertical" with a "buffer" of 3-4 terms. Their strategy wasn't revealed to be successful, indeed the experiment duration was above the average and they often argued about the other's scheduling, making a confused schedule. To explain the above cited statistics their position has been close to whiteboard because they worked on the scheduling based on the data already on the board, each course one by one, to fit a new course they had to check again all the current schedule. No similar procedure has been followed by experiments on the wall display.

Leadership dynamics appeared in experiments {1,4,6} with the addition of 7, whose collaboration was balanced with a heavy shift toward one participant. Overall, the majority of decisions were discussed together and there was not a dominant leader but a participant showed significant more decision making and conversation driving than the other. Interestingly all these experiments were grouped together when analyzing the verbal communication, showing a possible correlation between unbalanced data and leadership role taking. Interestingly, switching to visual attention, 4 and 6 are grouped together again as one of the most different in document attention and whiteboard attention, while 1 and 7 are grouped together in "average" difference in the same categories. This suggests that when leadership role takes place, differences in visual attention are more divergent in the whiteboard approach than the wall approach. But four cases does not constitute a statistically significant measure.

Experiments that showed a balanced collaboration with no evident leader are {2,3,10,12} with the addition of {5,8,9,11} where a small shift toward a participant happened. In the latter group a participant showed slightly more decision making and proposing than the other, while overall the collaboration style stayed flat.

5.9.10 Comparison of Results

Due to the many similarities of the works, some of my study results can be compared with the research conducted by Tantillo [34]. First I have to say that participants in his study were acting in a more similar way to the whiteboard approach due to the multi-touch input type. Furthermore, groups were formed of 4 people that could result more easily into leadership dynamics due to the higher number of participants. Also, the task was inherently different, as it involved two phases, one of which one was turn based and the other discussion and negotiation based.

His results though present similarities to mine: no significant improvement in performance nor in quality of work was found in the technological approach. But he states, similarly as I do, that improvements might be mitigated by the inherently complexity of the task. Also, participants claimed to prefer the software version to the traditional approach, similar to my experiments.

5.9.11 Signs of Frustration

While analyzing the videos I also paid attention to signs of frustration. In general no evident signs of frustration have been shown except in one experiment.

Only in experiment number 10, whiteboard approach. Many signs of frustration were shown after the first half of the duration. Most of the signs were due to a messy schedule and information displayed. This issue could have been solved by the wall approach since order, clearness and completeness of information is imposed by the system. This result is caused by their chosen approach to the problem. No leadership role was taken but division of the work was made. Subjects picked a couple of courses and they assigned one each to the schedule independently. This lead to some arguing about the schedule of the other participant.

CHAPTER 6

CONCLUSION

The purpose of this thesis is to give a contribute to the literature on human-computer interaction in the field of large format displays applied for project planning. At the same time it proposes four simple techniques to implement a general project planning application for large displays and it shows an innovative approach to enhance and automatize qualitative data analysis for similar studies.

I introduced the area of large format displays showing advantages as well as drawbacks and challenges. I gave a background on all topics and technologies employed in this thesis: SAGE2, project planning, collaboration, and I analyzed the reasons why the traditional way is still the most common approach.

I analyzed the state of art showing and discussing studies on large displays, on collaboration and lessons from the research. I also analyzed studies that adopted the same kind of data analysis (qualitative data analysis) employed in this thesis to find similarities and inspirations for the best approach.

I proposed in detail four simple techniques to implement a general project planning application for large displays. I described the details about the application design and implementation that was developed, with the same techniques, for the user study. Finally, I gave an overview of four different possibile implementations for different use cases, showing the versatility of such techniques.

I described the details of the user study that was conducted on collaboration. I introduced a new approach, explained in detail, to gather data from the study that helps the qualitative data analysis. I analyzed 12 groups of two people each, performing a planning task using the application developed on a large display or on a regular whiteboard. The goal was to create an admissible plan in a collaborative environment.

I analyzed human behaviors while performing such a task. Parameters of interest included visual attention, distance from the tool and sitting time, distance between subjects and verbal communication. Furthermore, other parameters were taken into account during the analysis: questionnaire answers, experiment duration, signs of frustration, and collaboration dynamics, such as leadership role taking.

Experiments show that the software and the offered technology is not missing any critical features and all the groups were able to complete the task on the wall display version of the experiment in no more time than the whiteboard approach. This proves that the software is in fact, a valid alternative to the traditional approach.

The improvement in performance in terms of time, though small, it's not sufficient to constitute a statistical evidence to state an improvement on task execution by the display version, nor in quality of result. The data gathered suggest that some behaviors might be slightly influenced by the employed technology, but again, the difference does not constitute a statistical evidence. Still, not identifying any decrease in performance or quality of results, is already a success. Furthermore, users enjoyed using the application and didn't find any significant difficulty.

CITED LITERATURE

1. Xiaojun, B., Seok-Hyung, B., and Ravin, B.: Effects of interior bezels of tiled-monitor large displays on visual search, tunnel steering, and target selection. Technical Report CHI 2010, Computer Science Department, University of Toronto, Atlanta, GA, USA, 2010.

2. Desney, S. T. and Mary, P. C.: Effects of visual separation and physical discontinuities when distributing information across multiple displays. Proc. Interact, 3, 2003.

3. Grudin, J.: Partitioning digital worlds: focal and peripheral awareness in multiple monitor use. Proceedings of the SIGCHI conference on Human factors in computing systems, pages 458–465, 2001.

4. Youn-ah, Kang & John, S.: Lightweight task/application performance using single versus multiple monitors: A comparative study. Technical report, School of Interactive Computing / GVU center Georgia Institute of Technology, Atlanta, GA, USA, 2008.

5. Desney, S. T., Mary, P. C., and George, G. R.: Large displays enhance optical flow cues and narrow the gender gap in 3-d virtual navigation. Human Factors: The Journal of the Human Factors and Ergonomics Society.

6. Desney, S. T., George, G. R., and Mary, C.: Exploring 3d navigation: Combining speed-coupled flying with orbiting. In Proceedings of the SIGCHI conference on Human factors in computing systems.

7. Christopher, A., Alex, E., and Chris, N.: Space to think: Large, high-resolution displays for sensemaking. In Proceedings of the SIGCHI conference on Human factors in computing systems.

8. Baudisch, P., Good, N., Bellotti, V., and Schraedley, P.: Keeping things in context: A comparative evaluation of focus plus context screens, overviews, and zooming. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '02, pages 259–266, New York, NY, USA, 2002. ACM.

9. Czerwinski, M., Robertson, G., Meyers, B., Smith, G., Robbins, D., and Tan, D.: Large display research overview. pages 69–74, 2006.

CITED LITERATURE (continued)

10. Moreland, K.: Redirecting research in large-format displays for visualization. Large Data Analysis and Visualization (LDAV), 2012 IEEE Symposium, pages 91–95.

11. Ni, T., Schmidt, G. S., Staadt, O. G., Livingston, M. A., Ball, R., and May, R.: A survey of large high-resolution display technologies, techniques, and applications. In IEEE Virtual Reality Conference (VR 2006), pages 223–236, March 2006.

12. Kruger, R., Carpendale, S., Scott, S. D., and Greenberg, S.: Roles of orientation in tabletop collaboration: Comprehension, coordination and communication. Computer Supported Cooperative Work (CSCW), 13(5):501–537, Dec 2004.

13. Scott, S. D., Carpendale, M. S. T., and Inkpen, K. M.: Territoriality in collaborative tabletop workspaces. In Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work, CSCW '04, pages 294–303, New York, NY, USA, 2004. ACM.

14. Ball, R. and North, C.: Effects of tiled high-resolution display on basic visualization and navigation tasks. In CHI '05 Extended Abstracts on Human Factors in Computing Systems, CHI EA '05, pages 1196–1199, New York, NY, USA, 2005. ACM.

15. Nutsi, A. and Koch, M.: Multi-user usability guidelines for interactive wall display applications. In Proceedings of the 4th International Symposium on Pervasive Displays, PerDis '15, pages 233–234, New York, NY, USA, 2015. ACM.

16. Marrinan, T., Aurisano, J., Nishimoto, A., Bharadwaj, K., Mateevitsi, V., Renambot, L., Long, L., Johnson, A., and Leigh, J.: Sage2: A new approach for data intensive collaboration using scalable resolution shared displays. In 2014 International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom)(COLLABORATECOM), volume 00, pages 177–186, Oct. 2015.

17. Serrador, P.: The impact of planning on project success-a literature review. 1, 01 2013.

18. Cicibas, H., Unal, O., and Demir, K. A.: A comparison of project management software tools (pmst). In Software Engineering Research and Practice, eds. H. R. Arabnia, H. Reza, L. Deligiannidis, J. J. Cuadrado-Gallego, V. Schmidt, and A. M. G. Solo, pages 560–565. CSREA Press, 2010.

19. Garey, M. R. and Johnson, D. S.: Computers and Intractability; A Guide to the Theory of NP-Completeness. New York, NY, USA, W. H. Freeman & Co., 1990.

CITED LITERATURE (continued)

20. Anderson, J. A.: Productivity, screens, and aspect ratios. Technical report, University of Utah, 2007.

21. North, C. A. A. E. C.: Toward characterizing the productivity benefits of very large displays. Human-Computer Interaction – INTERACT'03, pages 9–16.

22. Bi, X. and Balakrishnan, R.: Comparing usage of a large high-resolution display to single or dual desktop displays for daily work. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09, pages 1005–1014, New York, NY, USA, 2009. ACM.

23. Czerwinski, M., Smith, G., Regan, T., Meyers, B., Robertson, G., and Starkweather, G.: Toward characterizing the productivity benefits of very large displays. In (2003) Interact 2003. IOS Press, January 2003.

24. Bezerianos, A. and Isenberg, P.: Perception of visual variables on tiled wall-sized displays for information visualization applications. IEEE Transactions on Visualization and Computer Graphics, 18(12):2516–2525, Dec 2012.

25. Ragan, E. D., Endert, A., Bowman, D. A., and Quek, F.: How spatial layout, interactivity, and persistent visibility affect learning with large displays. In Proceedings of the International Working Conference on Advanced Visual Interfaces, AVI '12, pages 91–98, New York, NY, USA, 2012. ACM.

26. Agostini, A. and Biase, E. D.: Large multitouch screens to enhance collaboration in the classroom of the 21st century: an italian experiment. IxD&A, 15:40–56, 2012.

27. Prensky, M.: Digital natives, digital immigrants part 1. On the Horizon, 9(5):1–6, 2001.

28. Isenberg, P., Fisher, D., Paul, S. A., Morris, M. R., Inkpen, K., and Czerwinski, M.: Co-located collaborative visual analytics around a tabletop display. IEEE Transactions on Visualization and Computer Graphics, 18(5):689–702, May 2012.

29. Rogers, Y. and Lindley, S. E.: Collaborating around vertical and horizontal large interactive displays: which way is best? Interacting with Computers, 16:1133–1152, 2004.

30. Vogt, K., Bradel, L., Andrews, C., North, C., Endert, A., and Hutchings, D.: Co-located collaborative sensemaking on a large high-resolution display with multiple input devices. In Human-Computer Interaction – INTERACT 2011, eds. P. Campos, N.

CITED LITERATURE (continued)

Graham, J. Jorge, N. Nunes, P. Palanque, and M. Winckler, pages 589–604, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

31. Nolte, A., A Brown, R., Poppe, E., and Anslow, C.: Exploring collaborative modeling of business processes on large interactive touch display walls. 01 2015.

32. Jakobsen, M. R. and HornbAEk, K.: Up close and personal: Collaborative work on a high-resolution multitouch wall display. ACM Trans. Comput.-Hum. Interact., 21(2):11:1–11:34, February 2014.

33. Pellolio, F.: Sageboard: a whiteboard for large multitouch displays. 2016.

34. Tantillo, D.: Large multi-touch vertical displays in multi-user competitive tasks. 2016.

35. Doshi, V., Tuteja, S., Bharadwaj, K., Tantillo, D., Marrinan, T., Patton, J., and Marai, G. E.: Stickyschedule: An interactive multi-user application for conference scheduling on large-scale shared displays. In Proceedings of the 6th ACM International Symposium on Pervasive Displays, PerDis '17, pages 2:1–2:7, New York, NY, USA, 2017. ACM.

36. Tang, A., Tory, M., Po, B., Neumann, P., and Carpendale, S.: Collaborative coupling over tabletop displays. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '06, pages 1181–1190, New York, NY, USA, 2006. ACM.

37. Knudsen, S., Jakobsen, M. R., and Hornbæk, K.: An exploratory study of how abundant display space may support data analysis. In Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design, NordiCHI '12, pages 558–567, New York, NY, USA, 2012. ACM.

38. Patrick, E., Cosgrove, D., Slavkovic, A., Rode, J. A., Verratti, T., and Chiselko, G.: Using a large projection screen as an alternative to head-mounted displays for virtual environments. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '00, pages 478–485, New York, NY, USA, 2000. ACM.

VITA

NAME	Tommaso Massari
EDUCATION	
	Master of Science in “Computer Science, University of Illinois at Chicago, Aug 2017, USA
	Specialization Degree in “Computer Science and Engineering”, Sep 2016, Polytechnic of Milan, Italy
	Bachelor’s Degree in Computer Science and Engineering, Sep 2013, Polytechnic of Milan, Italy
LANGUAGE SKILLS	
Italian	Native speaker
English	Full working proficiency
	2017 - Toefl iBT examination (92/120)
	A.Y. 2017/18 One Year of study abroad in Chicago, Illinois
SCHOLARSHIPS	
Spring 2017	Research Assistantship (RA) position (20 hours/week) with full tuition waiver plus monthly stipend
Fall 2018	RA position (20 hours/week) with full tuition waiver plus monthly stipend
TECHNICAL SKILLS	
Basic level	C++, Python, HTML, CSS,
Average level	Java, C#, SQL, Assembly, Maya, Matlab
Advanced level	C, JavaScript, Unity3D, Arduino
WORK EXPERIENCE AND PROJECTS	
Jan 2018 - Dec 2018	Research Assistantship (RA) position at UIC

VITA (continued)

Aug 2018 - Dec 2018	Research Assistantship (RA) position at University of Chicago