Data Visualization in the Data Center Infrastructure Management field

Joan Maller Roig

Bachelor Thesis
Specialization in Computing

Director: Antonio Suárez
Tutor: Dr. Pere-Pau Vázquez
GEP Tutor: Joan Carles Gil

22nd October 2021

UPC
UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona
FIB

Abstract

The Digital Revolution has been characterized by a widespread use of the Internet and the generation of huge quantities of data. The computation and analysis of this data is not always automatic. It often needs human help to first decide the purpose or goal of the task-at-hand and then select the right features from the data that will help achieve that goal. Data visualization aims to facilitate this decision-making process.

Such vast amounts of data are stored and processed in data centers which require the proper maintenance of their physical ecosystems (i.e., their space, electricity, temperature, and data security), which is mostly done though Data Center Infrastructure Management (DCIM) systems.

Research in this field has mostly focused on software solutions to manage these types of systems and hasn't focused as much on data visualization. The purpose of this study is to contribute in this field by proposing a data visualization solution that aims at helping users of DCIM systems to better analyze and interpret the critical data (i.e. resources and alarms). This will allow users not only to make better decisions on forecasting and planning and to manage these systems more safely and efficiently.

To do so, this solution is designed using multiple task-based data visualizations, a custom data pipeline and a tailored website from back-end to front-end. The latter has the main focus of the project, since it is the final user interface experience, but every step of the solution counts and is given equal importance to achieve a solid solution.

Resum

La Revolució Digital s'ha caracteritzat per un ús generalitzat d'Internet i la generació de grans quantitats de dades. El càlcul i l'anàlisi d'aquestes dades, però, no sempre són automàtics. Sovint cal l'ajuda humana per decidir primer el propòsit o l'objectiu de la tasca en curs, i després seleccionar les dades adequades que ajudaran a assolir aquest objectiu. La visualització de dades té com a propòsit facilitar el procés de presa de decisions.

Aquestes grans quantitats de dades s'emmagatzemen i es processen en centres de dades que requereixen el manteniment adequat dels seus ecosistemes físics (el seu espai, electricitat, temperatura i seguretat de les dades), que es fa principalment mitjançant programaris anomenats sistemes de gestió d'infraestructures de centres de dades (DCIM).

La investigació en aquest camp s'ha centrat sobretot en solucions de programari per gestionar aquest tipus de sistemes i en canvi no s'ha centrat tant en la visualització de dades. L'objectiu d'aquest estudi és contribuir en aquest camp proposant una solució de visualització de dades que té com a objectiu ajudar els usuaris de sistemes DCIM a analitzar i interpretar millor les dades crítiques (principalment recursos i alarmes). Això permetrà als usuaris prendre millors decisions, predir, planificar i gestionar aquests sistemes de manera segura i eficient.

Aquesta solució ha estat dissenyada a través de visualitzacions definides a partir de tasques, un camí propi de tractament de les dades i una web a mida de l'usuari. Aquesta última, és la part més visible del projecte, ja que és la interfície final a nivell d'usuari, tanmateix, cada part de la solució té una rellevància en el projecte.

Resumen

La Revolución Digital se ha caracterizado por un uso generalizado de Internet y la generación de gran cantidad de datos. El cálculo y el análisis de estos datos, sin embargo, no siempre es automático. A menudo es necesaria la colaboración humana para decidir primero el propósito o el objetivo de la tarea en curso, y luego seleccionar los datos adecuados que ayudarán a alcanzar este objetivo. La visualización de datos tiene como propósito facilitar este proceso de toma de decisiones.

Esta gran cantidad de datos se almacena y se procesa en centros de datos que requieren el mantenimiento adecuado de sus ecosistemas físicos (su espacio, electricidad, temperatura y seguridad de los datos), lo que se hace principalmente mediante un software llamado sistema de gestión de infraestructuras de centros de datos (DCIM).

La investigación en este campo se ha centrado sobre todo en soluciones de software para gestionar este tipo de sistemas y en cambio no se ha centrado tanto en la visualización de datos. El objetivo de este estudio es contribuir en este campo proponiendo una solución de visualización de datos que tiene como objetivo ayudar a los usuarios de sistemas DCIM a analizar e interpretar mejor los datos críticos (principalmente recursos y alarmas). Esto permitirá a los usuarios tomar mejores decisiones sobre previsiones y planificación, gestionando estos sistemas de forma más segura y eficiente.

Esta solución ha sido diseñada mediante visualizaciones definidas a partir de tareas, un camino de tratamiento de datos propio y una web a medida del usuario. Ésta última, es la parte más visible del proyecto, ya que es la interfaz final a nivel de usuario, sin embargo, cada parte de la solución tiene una relevancia en el proyecto.

Contents

List of Figures

List of Tables

1 Introduction and contextualization

Once data was managed in notebooks, from small businesses to bigger ones, where there was maybe an accountant doing the numbers or keeping the stock. And decisions were taken eyeballing the data. The appearance of computers, and subsequently, the Internet, has led to a massive growth in the size of data. And the possibility to analyze it or predict the future with AI, has contributed to what we now call the second information revolution [7].

Data has taken a lot of importance recently through the many uses it can have, and it may be the most valuable asset in a company. Added to the use of Internet and ecommerce, the use of dedicated computers storing the data has grown exponentially, and nowadays, many big businesses have a data center. The origins of the data centers are in the ENIAC computers used by governments that required a special environment ensuring a large space, maintenance and data security [8]. Now, they also occupy a spacial environment, where they have to be maintained and taken care of to ensure data security.

This is the work of Data Center Infrastructure Management (DCIM), a smart solution to manage a physical ecosystem with the knowledge of how to reduce operational costs and make critical infrastructure safe and running. Nobody wants a web down or an entire application down. For this reason, managing data centers correctly, and being more efficient, thus saving money, has been a field of research the last few years. In this project, we are using the power of data visualization to ease the decision-making process made by humans in managing a DCIM system.

1.1 Context

This is a Bachelor Thesis from the Computer Science degree, where my specialization is in Computing, done in the Facultat d'Informàtica de Barcelona (FIB) faculty of the Universitat Politècnica de Catalunya (UPC) university. This thesis is directed by Antonio Suárez, the head of the Innovation team at Bjumper, a DCIM centered company. And tutored by Dr. Pere-Pau Vázquez, Associate Professor of the Department of Computer Science, at UPC, whose research interests include data visualization.

1.2 Concepts

It is important to be familiarized with these concepts to better understand the following parts. A brief description will be done, for getting a technical knowledge of the project.

1.2.1 DCIM - Data Center Infrastructure Management

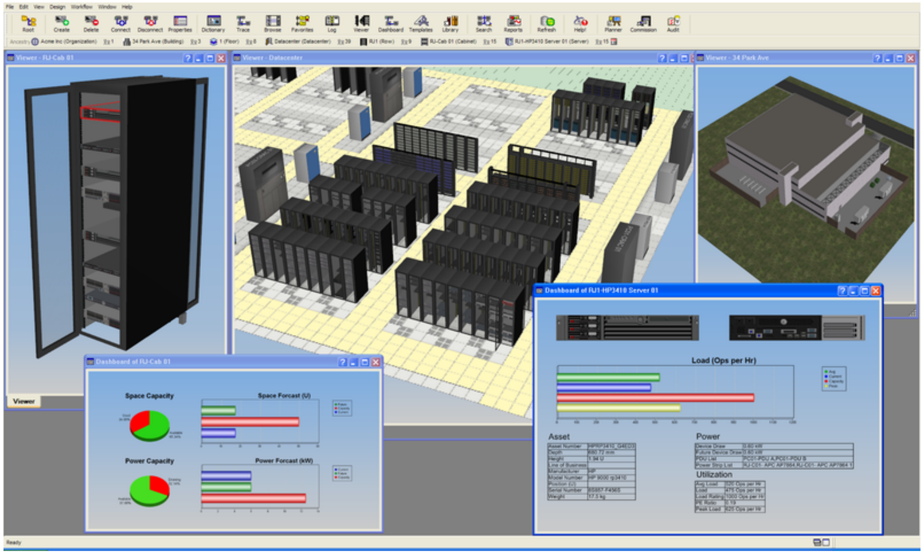
A DCIM solution integrates the physical infrastructure (facilities) with the interconnectivity between assets, systems, and workflows. The value stack of DCIM is really big:

- **Operations management**
  Real-time monitoring of physical ecosystem and devices or day-to-day operational visibility - both IT and facilities.
- **Resource management (power, space, network, cooling)**
  Resource consumption, optimizing the efficiency and use of space, power, cooling and network ports. Identify possible issues and improvements on any of the resources.
- **Change management**
  Planning, execution, and verification of infrastructure changes. Select best locations for moves, adds, an changes based on future resource availability. Avoid stranded power, eliminate hot spots. Make the right change at the right time.
- **Asset and connectivity management**
  Tracking, management, and utilization of assets throughout their entire lifecycles. Including port-to-port network connectivity.
- **Availability management**
  Identification and resolution of technology and process issues that could impact uptime. Find and mitigate sources of risk - predictive modeling.
- **Capacity planning and demand management:**
  Forecasting, planning, and allocation of resources over time, tracking growth of IT portfolio and identifying future points when new data center investments will be required.

Understanding DCIM as a whole, capable of doing all the above, is primordial, with the idea that some of them will be better in some areas than the others.

Examples

Here are some DCIM system examples that the company works with, the first one being iTRACS (figure 1), developed by Commscope and the second system divided in two applications being ITA (figure 2) and DCE (figure 4), both from Schneider Electric.


File Edit View Design Workflow Window Help
Root Create Delete Connect Disconnect Properties Dictionary Trace Browse Favorites Log Viewer Dashboard Templates Library Search Reports Refresh Help! Planner Commission Audit
Ancestry Acme Inc (Organization) 34 Park Ave (Building) 1 (Floor) Datacenter (Datacenter) RJ1 (Row) RJ-Cab 01 (Cabinet) RJ1-HP3410 Server 01 (Server)
Viewer - RJ-Cab 01
Viewer - Datacenter
Viewer - 34 Park Ave
Viewer
Dashboard of RJ-Cab 01
Space Capacity
Space Forcast (U)
Power Capacity
Power Forcast (kW)
Dashboard of RJ1-HP3410 Server 01
Load (Ops per Hr)
Avg
Current
Capacity
Peak
0 100 200 300 400 500 600 700 800 900 1000 1100 1200
Ready

Asset

Asset Number	HPRP3410_G4EO3
Depth	680.72 mm
Height	1.94 U
Line of Business	
Manufacturer	HP
Model Number	HP 9000 rp3410
Position (U)	
Serial Number	8S857-F4565
Weight	17.5 kg

Power

Device Draw	0.60 kW
Future Device Draw	0.60 kW
PDU List	PC01-PDU A,PC01-PDU B
Power Strip List	RJ-C01- APC AP7864,RJ-C01- APC AP7864 1

Utilization

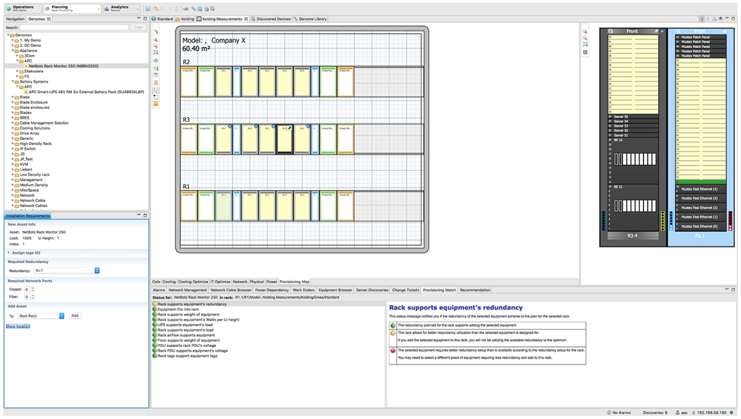
Avg Load	520 Ops per Hr
Load	475 Ops per Hr
Load Rating	1000 Ops per Hr
PE Ratio	0.19
Peak Load	625 Ops per Hr

Figure 1: iTRACS, the DCIM application developed by Commscope, focused on 3d visualization. Source: iTRACS [1]

iTRACS is a complete program having both facilities and alarms in the same application, and with a 3d-type visualization as the main navigator. But with other views as a hierarchy view, or a dashboard. To make it work, it is needed a team of iTRACS experts, that make a copy of your real data center into the virtual space, taking care of every equipment layout and correctly labeling every cable, sensor and possible facility in the data center.

The main advantages are the clarity and realism of the visualization, detailed to the cable scale, and clearly the completeness of the application. When we hover the mouse on a facility, for example a rack, it will tell us all the properties, real-time measurements and even the path that every cable that is connected follows. The main disadvantage is the speed, due to the many graphic needs that it has and quantity of data that loads for every view, making it more or less "fluent" only in powerful computers equipped with graphic cards.

Schneider Electric dedicates a branch to the data center management, that divides its software in two principal applications, the first one dedicated to the facilities and the second to the alarms. It has also lately focused on the web version, adding the 3d version and an interesting room temperature heat diffusion map.



New Asset Info

Asset: NetBotz Rack Monitor 250

Load: 1000 U-Height: 1

Inlets: 1

Assign tags (0)

Required Redundancy

Redundancy: N+1

Required Network Ports

Copper: 0

Fiber: 0

Add Asset

To: Best Rack Add

Show location

Status for: NetBotz Rack Monitor 250 **in rack:** R1.1/R1/Model: /Kolding Measurements/Kolding/Emea/Standard

- Rack supports equipment's redundancy
- Equipment fits into rack
- Rack supports weight of equipment
- Rack supports equipment's Watts per U-height
- UPS supports equipment's load
- Rack supports equipment's load
- Rack airflow supports equipment
- Floor supports weight of equipment
- PDU supports rack PDU's voltage
- Rack PDU supports equipment's voltage
- Rack tags support equipment tags

Rack supports equipment's redundancy

This status message notifies you if the redundancy of the selected equipment adheres to the plan for the selected rack.

🟢	The redundancy planned for the rack supports adding the selected equipment.
🟡	The rack allows for better redundancy utilization than the selected equipment is designed for. If you add the selected equipment to this rack, you will not be utilizing the available redundancy to the optimum.
🔴	The selected equipment requires better redundancy setup than is available according to the redundancy setup for the rack. You may need to select a different piece of equipment requiring less redundancy and add to this rack.

Figure 2: ITA: ITAdvisor, an application of the DCIM solution by Schneider, focused on a simple visualization of the facilities floor plan and a panel for rack front and rear view. Source: Schneider Electric [2]

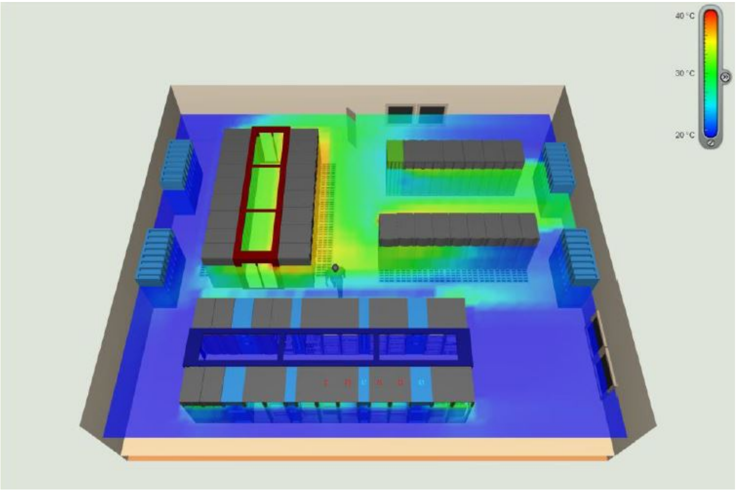
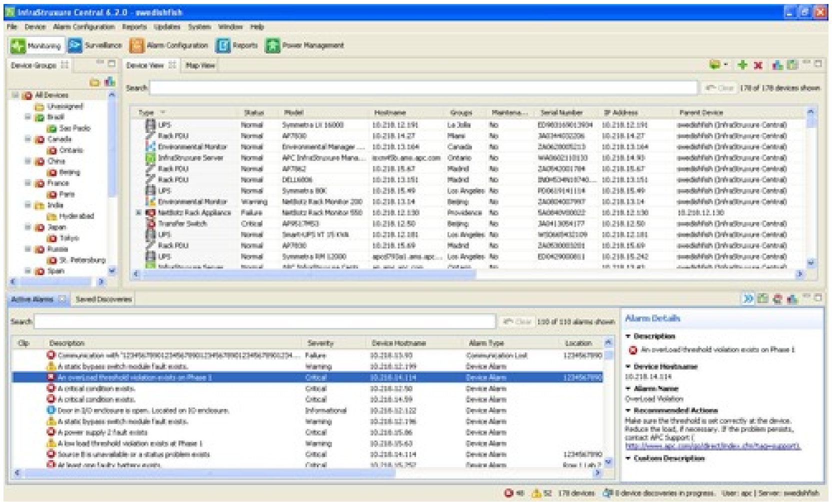


Figure 3: DCO: Data Center Operations, a cloud-based application of the DCIM solution by Schneider, with a 3d visualization of room temperature. Source: Schneider Electric [3]

ITA is the facilities application, giving us a view of the floor plans of the data center, and being able to see a front and rear rack view panel upon clicking on them. DCO is the name of the web-version of the application.

DCE is where all the sensors and IP registered devices are shown and consequently, all the live data and alarms appear, being able to manage the different data center incidents from there. There are 4 different types of alarms, as we see from the color of the symbols: Critical, Failure, Warning and Informational. It is used to monitor the data center on a real-time basis.



Type	Status	Model	Hostname	Groups	Maintena...	Serial Number	IP Address	Parent Device
UPS	Normal	Symmetra LX 16000	10.218.12.191	La Jolla	No	ED0831690135904	10.218.12.191	swedishfish (InfraStruxure Central)
Rack PDU	Normal	AP7930	10.218.14.27	Miami	No	3A0344032206	10.218.14.27	swedishfish (InfraStruxure Central)
Environmental Monitor	Normal	Environmental Manager ...	10.218.13.164	Canada	No	ZA0628005213	10.218.13.164	swedishfish (InfraStruxure Central)
InfraStruxure Server	Normal	APC InfraStruxure Mana...	isxm45b.ams.apc.com	Ontario	No	WA0602110130	10.218.14.93	swedishfish (InfraStruxure Central)
Rack PDU	Normal	AP7862	10.218.15.67	Madrid	No	ZA0542001784	10.218.15.67	swedishfish (InfraStruxure Central)
Rack PDU	Normal	DELL6006	10.218.13.151	Madrid	No	INDH534N18740...	10.218.13.151	swedishfish (InfraStruxure Central)
UPS	Normal	Symmetra 80K	10.218.15.49	Los Angeles	No	PD0619141114	10.218.15.49	swedishfish (InfraStruxure Central)
Environmental Monitor	Warning	NetBotz Rack Monitor 200	10.218.13.14	Beijing	No	ZA0804007997	10.218.13.14	swedishfish (InfraStruxure Central)
NetBotz Rack Appliance	Failure	NetBotz Rack Monitor 550	10.218.12.130	Providence	No	5A0840V00022	10.218.12.130	10.218.12.130
Transfer Switch	Critical	AP9517M53	10.218.12.50	Beijing	No	3A0413054177	10.218.12.50	swedishfish (InfraStruxure Central)
UPS	Normal	Smart-UPS VT 15 KVA	10.218.12.181	Los Angeles	No	WS0605432109	10.218.12.181	swedishfish (InfraStruxure Central)
Rack PDU	Normal	AP7930	10.218.15.69	Madrid	No	ZA0530003201	10.218.15.69	swedishfish (InfraStruxure Central)
UPS	Normal	Symmetra RM 12000	apcd793a1.ams.apc...	Los Angeles	No	ED0429000811	10.218.15.242	swedishfish (InfraStruxure Central)

Clip	Description	Severity	Device Hostname	Alarm Type	Location
	Communication with "12345678901234567890123456789012345678901234...	Failure	10.218.15.93	Communication Lost	1234567890
	A static bypass switch module fault exists.	Warning	10.218.12.199	Device Alarm	
	An overLoad threshold violation exists on Phase 1	Critical	10.218.14.114	Device Alarm	1234567890
	A critical condition exists.	Critical	10.218.12.50	Device Alarm	
	A critical condition exists.	Critical	10.218.14.59	Device Alarm	
	Door in I/O enclosure is open. Located on IO enclosure.	Informational	10.218.12.122	Device Alarm	
	A static bypass switch module fault exists.	Warning	10.218.12.196	Device Alarm	
	A power supply 2 fault exists	Critical	10.218.15.86	Device Alarm	
	A low load threshold violation exists at Phase 1	Warning	10.218.15.63	Device Alarm	
	Source B is unavailable or a status problem exists	Critical	10.218.14.114	Device Alarm	1234567890
	At least one faulty battery exists.	Critical	10.218.15.252	Device Alarm	Row 1 Lab 2

Alarm Details

▾ Description

An overLoad threshold violation exists on Phase 1

▾ Device Hostname

10.218.14.114

▾ Alarm Name

OverLoad Violation

▾ Recommended Actions

Make sure the threshold is set correctly at the device. Reduce the load, if necessary. If the problem persists, contact APC Support (http://www.apc.com/go/direct/index.cfm?tag=support).

▾ Custom Description

46 52 178 devices 0 device discoveries in progress. User: apc | Server: swedishfish

Figure 4: DCE: Data Center Expert, an application of the DCIM solution by Schneider, with all the IP-connected device information and alarms. Source: Schneider Electric [4]

Types of data centers

As a company grows, it needs a place to store and maintain the data, normally as a form of server or collection of them. At this moment, the company must choose if it is better to have the store and maintain the data in the company or leave the task to a private company.

If the company doesn't have the resources, it may be easier to get the job done by another business. You can rent a server or a collection of servers in a Colocation data center, also referred as Co-Lo, paying a price monthly for their use, being it much less than buying, building and maintain your own data center.

Another alternatives are Amazon S3, Microsoft's Azure and Google cloud, being three popular among different cloud solutions, the main advantages being the scalability and easiness of use for the low-skilled users.

It is also important to look for uptime and security of data when deciding for one service or another, to avoid possible losses of data or cybersecurity attacks.

The uptime is the percentage of time the server (containing the website, service or data) is accessible, it is written in the SLA (Service Level Agreement), and is also present in some data center certifications. One of the most popular of them being the TIER classification, given by the Uptime Institute, that goes from 1 to 4 and 99.675% to 99.995%. There are many other things that are taken into account for a Tier certification, like redundancy (when a power outlet fails, you have a backup power line) and power protection time (first minutes by batteries and then usually by an electric generator).

Cybersecurity has taken a lot of relevance lately, due to the appearance of multiple cyberattacks. An example of it is the late ransomware attack to the UAB university, affecting more than 650 thousand files, and asking for 3 million euros in return.

1.2.2 Data visualization

These multiple DCIM systems, product huge quantities of data, analyzing the multiple sources and taking automatic action based on the data. For example, when an uprise of room temperature happens, due to the heat of data centers, it can email the data center manager to be attentive. But these actions are not always automatic, sometimes human decisions must be made to approach the problem, setting goals or features from the dataset.

However, to take a decision based on data, we must navigate a large dataset of information, impossible for the human to catch with a single view. Data visualization allows us to do that, selecting or summarizing the dataset, and then using vision as a perfect information processor. Visualization, as the name implies, is based on exploiting the human visual system as a means of communication. [5] With the wide bandwidth of vision, we can detect multiple information that exists in our eye space and detect a red point in a cloud of gray ones. The understanding of perception(link a perception) gives us knowledge to design better data visualizations.

Definition

"Data visualization is the representation and presentation of data to facilitate understanding." [9]

The word **representation** is deliberately positioned by Andy Kirk near the front of the definition because he says it is the quintessential activity of data visualization design. Representation concerns the choices we make about the form in which the data will be visually portrayed: in layout terms, what possible charts will you use to exploit the brain's visual perception capabilities most effectively.

When data visualizers want to create a visualization, they will be representing the data through combinations of marks and attributes. Marks are the points, lines, and areas. Attributes are the appearance properties of these marks, like the size, color, and position. The mix of these marks and their attributes form the skeleton of the chart.

The **presentation** of data concerns for all the other visible design decisions that make up the overall visualization anatomy. This includes possible applications of interactivity, features of annotation, color usage and the composition of your work.

Perception

Perception theory is to be taken into account when designing an efficient data vis, understanding how human vision catches an image, to get foundations for the applied science of data visualizations.

I have to thank my tutor for his recommendation of the book by Colin Ware named Information Visualization [10] and its focus on perception has helped me gain an insight of how light, color, Gestalt laws, Depth Cue and Visual thinking algorithms work.

Storytelling

Important stories live in our data, and data visualization is a powerful means to discover and understand these stories, and then to present them to others. As the saying goes, "a picture is worth a thousand words" - often more - but only when the story is best told graphically rather than verbally and the picture is well-designed. Although data visualization usually features relationships between quantitative values, it can also display relationships that are not quantitative in nature.

Dataset availability

When we think of the dataset, we always think the dataset is finished and ready to be consumed, as a static file. However, some datasets are available in a dynamic streams, and the information arrives as we are looking at the data visualization, for example the stock market.

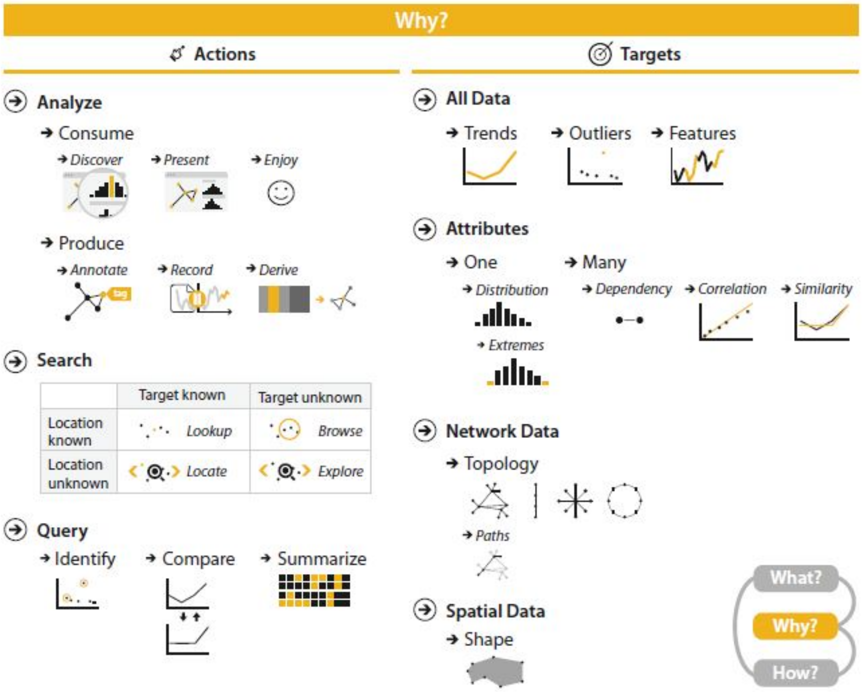
➔ Static
➔ Dynamic

Figure 5: Dataset availability. Source: [5]

When we are dealing with dynamic datasets, we can add, remove or change current data as we receive the new one, which adds a layer of complexity that we have to deal with. For example, downsampling is a method for treating new data, that we will explain later in the development section.

Task asbtraction

Why a vis tool is going to be used, is an essential question to make ourselves as the first step in designing a data visualization. Figure 6 summarizes this step. It is divided between actions and targets.



	Target known	Target unknown
Location known	Lookup	Browse
Location unknown	Locate	Explore

Figure 6: Task Abstraction. Source: [5]

The actions we may use the data visualization for are Analyzing, Searching and Querying data. Discovery is when we are searching for a hypothesis or validating it, finding new knowledge that was previously unknown. Present is the common communication of information, through storytelling, in the context of decision-making, planning and forecasting. This is and Discovery are the actions in the Analyzing type we are going to use data vis for.

We may use data vis because we want to search for data, as a mid-level goal, and we are going to do in the Search type actions. and also the Query types for a low-level user, accompanied by interactivity. Trend to see, for example, the space that is occupied in the data center and forecast when we need to buy more facilities or rooms. And we might look at outliers or features task dependent, and all the attributes listed in the figure. Finally, we could look for trees based on the hierarchical anatomy of a data center, paths to look for dependencies, and shapes of rack templates to draw a floor plan on spatial data.

1.2.3 Data warehouse

A Data warehouse is an integrated collection of data, from different sources, used for analysis or reporting. Many products build a data warehouse of its own, for quicker access, while in the background it can be updated with multiple back-end operations. One of the other reasons is to restructure data, so the new construction makes more sense to the use the company will do to it.

When we are working with APIs and source DBs from DCIM software products, we find that a good deal of information is redundant, which brings us to building a new data warehouse, adding also proprietary information tailored to the end product.

It is not easy to build a data warehouse, firstly we have to find the data we want to collect from the sources we have access to, later we may want to clean our data, running operations through the data, and then use a data warehouse architecture[11] of our choosing to load the data on.

Moreover, we need to have an eye into the future, to make our data warehouse functional today and tomorrow. Particularly looking for new technologies in the market, searching for data sources, and building a solution that is both scalable and flexible.

1.3 Problem to be solved

The problem of DCIMs is that they have a huge task and many areas to develop. Consequently, it is difficult to put a lot of options in the software while also making it easy and intuitive. In the field there are lots, but the most famous are ITRACS by Commscope, DCE/ITA by Schneider Electric and SunbirdDCIM. Most of them have a web version for accessing the software, or a light-version of it. And some are looking into smartphone variants.

The advantages of having a web version are clear, access from anywhere and no need of local installation of the software. The disadvantages are also remarkable today, long loading times because of the size of the data displayed, and difficulty to find things.

One of the fields where the web version has missed a spot is the data visualization field. All of them have tried a general overview / Dashboard of the data center, but most have failed at doing so in a clear and useful way. Here is where we get involved, trying to come up with a new way to view DCIM data.

1.4 Stakeholders

Lots of people are involved in this project, we have two types of stakeholders, primary and secondary. The primary stakeholders, who are directly affected by the project, in a beneficial or detrimental way. My project director, the company head of Innovation team, Antonio Suárez. Moreover, all the Bjumper company workers, investors or anyone directly affected in an economic relation.

The secondary stakeholders, are the people who are not directly related to the company, but are in the field and may be affected by our project. The DCIM field, and many of our competition companies, could be affected by our result.

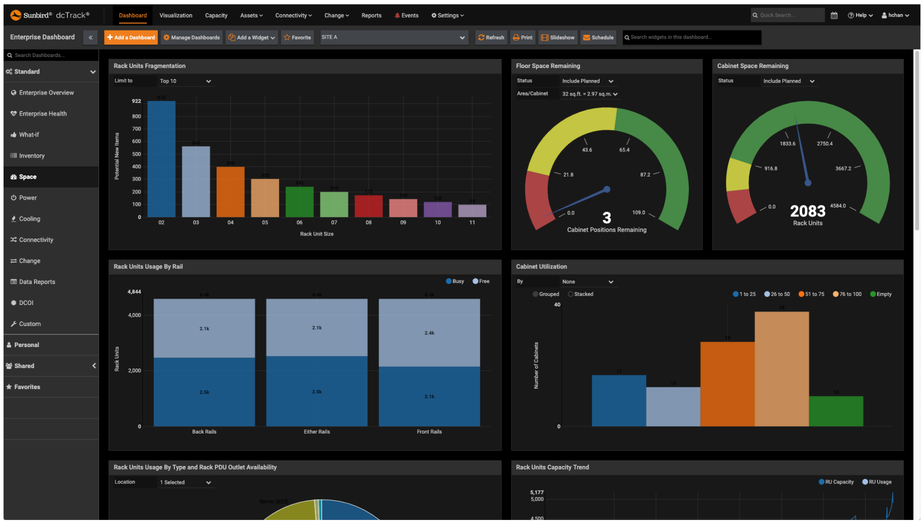
2 Justification

In this part, I will discuss the state-of-the-art and make a selection of the best technology for our project. Firstly, through the previous studies made to find the different solutions existing in the market. And secondly, a justification of our choice.

To better define the technologies used in the project, I have been doing some individual pre-development tasks, research of the state-of-the-art, and testing some technologies to see if they are adequate for our tasks. If we search for DCIM technologies, and look at their websites, we see a common layout called dashboard. A dashboard is a graphical user interface, that gives us a current summary of our data, normally using KPI(Key Performance Indicators). We find that Intel DCM, iTRACS, Schneider Electric, SunbirdDCIM and OpenDCIM have different types of dashboards.

The most convincing of them all is the SunbirdDCIM[6], which you can see in Figure 7 added below. We can see a basic dashboard website with multiple configuration options, many tabs in the top of the page and a menu in the left, a search bar and many configuration options.

It looks like it is based in Grafana, although the data visualization plots are slightly different in design. Grafana is a monitoring solution or dashboard software, once you select the data source, then you query the data, and finally you choose a graphing solution that better fits you. It is highly customizable, and has many plugins for connecting to different data sources and to plot different data visualizations. An interesting topic about Grafana[12] is the connection with Graphite and InfluxDB, modules for TSDB(Times Series Database)[13], a database that stores a set of values for a specific timestamp, that are evidently popular in the monitoring world, using real-time data.



SITE A

Limit to Top 10

Status Include Planned

Area/Cabinet 32 sq.ft. = 2.97 sq.m.

Status Include Planned

By None

Location 1 Selected

Figure 7: SunbirdDCIM example capacity dashboard. Source: SunbirdDCIM [6]

Alternatives to Grafana are Zabbix[14], Kibana or Tableau. The first one is a little outdated in terms of design, the second is more thought for log-based data sources, but has an excellent Elasticsearch module that makes really efficient searches on data, so it might be a pretty good alternative. Tableau is a very well-known but commercial application, and my company has made the choice to work with Open-source only applications for this project, so we are not going to consider it. Qlik, PowerBI, Sisense and much others are possible alternatives but also have the drawback that they are paid.

This is what has been already created, but we want to do better, and we have done some research in other ways of doing. The main idea behind it is a data pipeline, from collecting data, message breaking (queues), a proprietary back-end server, using a designed data warehouse and a proprietary front-end using multiple data visualizations from different libraries depending on the data we have. The back-end will be in Node.js and the front-end in Angular, using Vega, Vega-lite and D3.js data visualization libraries.

The final version could include a hybrid of the existing technologies with our own due to the short time of the project.

3 Scope

This section explains where the project begins and where it ends, its main objectives, possible risks that we can find in the process, and methodology to follow.

3.1 Objectives

To have success on the project, we have to visualize a series of achievable objectives.

1. **Research current DCIM monitoring technologies** First, research is crucial in the design of our technology, so searching for the current market solutions and alternatives, is a must.
2. **Define achievable tasks** Defining achievable tasks, is primordial before searching the data. Before creating a data visualization, we need to define what do we need that data for. With this step, we set our goal clearly and it will define the rest of the process.
3. **Find data sources** We have to find the information we want to display in order to make our application functional.
4. **Build data warehouse** A data warehouse that integrates many data sources, has to be designed in a well-thought architecture, making it easy maintainable and scalable.
5. **Develop a data visualization website** The last objective makes up to the end user interface, maybe the one which will have more importance, so a good design, from back-end to front-end will be key.
6. **Make usability tests** During the multiple stages of this project, we have to make usability tests, the reason to make them is to avoid wrong software choices. It is difficult due to the short time of the project, but can be achieved.

3.2 Requirements

The following requirements are needed to ensure the quality of the project.

- Design and develop an efficient data warehouse.
- Design an intuitive and clean website front-end.
- Build an efficient and fast back-end.

3.3 Obstacles and risks

It is possible that we find the following obstacles and risks. To prevent them, we are going to firstly expect them, and secondly take part in avoiding them.

- **Design errors** The first mistakes we have to avoid are design errors, because are the ones we are going to do first. If we solve them, we can enjoy a better architecture, efficiency and function.
- **Implementation errors** It is impossible that these implementation errors do not happen, but we have to be watchful and detect them as soon as possible, and try to solve it with our teammates or online. Debugging can help us a lot in this part.
- **Technical problems** We need to be careful with the technical problems and make backups of data, tools and local configurations we make. Server problems have occurred, and we are going to prevent them from ruining our work.
- **Temporal** Maybe the most possible of them all, the no completion of objectives in time is the major risk for us. To avoid it, we have to think ahead, learning to adapt tasks to the rhythm of the team, bettering the spirit of the teammates. Time is the line that makes the project be finished or not, so we have to adapt and do a minimum product before the end of it, so later we can upgrade it into a better product and still be in time.

3.4 Methodology

The first weeks were spent doing research and testing, to see the viability of the project. This is an individual task, later a group of 2 people joined the team to do a training and development part.

We are going to use the Agile methodology, that uses sprints to complete tasks in a short period of time. So, dividing the problem into sub-tasks is decisive to make the project possible. Weekly goals are set, and sprints may take 1 or 2 weeks to complete. Sprints can be very powerful to dedicate all of our efforts, as individuals or team, to a specific task, and take us to the next level of expertise in that task, surrounding us by a lot of research and trial and error. After a sprint has been done, we are going to dedicate some time to document the work done. And making a usability test directly with our superior or with people form outside the company. For easing the work to other employees, and to our future selves come back anytime we want to the sprint that we made in a specific time.

3.4.1 Tools

Our company uses the **Microsoft 365** package, so we will be taking advantage of the license, using **SharePoint, Teams, Office** and **OneDrive** to centralize document storing and sharing. **GitHub** will be used for version control, the repository will be private for the Bjumper team. We use **Odoo**, a suite for business management, that includes a module for goal setting and time management, to see whether you spent the time you planned or more. And on an individual part, **Trello**, a kanban styled planning website, based in subtasks as cards that you divide in different time or priority boards.

3.4.2 Validation

Validation of the project is being done by scheduled weekly reunions, through a follow-up session, to see if the weekly goals have been accomplished or not. Hence, giving feedback to the director, whether the goals had been set correctly. In addition, these reunions help us coordinate better, and think of next week goals.

It is essential that usability tests are made to validate the effectiveness of each data visualization, so that it meets the requirements defined by the tasks that it has to solve. These have been done at different levels of skill, from data vis designers to technical data center managers including low-skilled users depending too on the user of the data visualization. This will be developed in the validation section 6.1 of the results.

4 Schedule

The total number of hours I will be working with Bjumper, from April 26th to September 3rd, are 752 hours. The total number of hours dedicated to this project are the most part of my time. First of all, I have to define the tasks ensuring the completion of the project, it is important to define tasks correctly, and visualize the viability of the project and possible obstacles or risks that run against it.

Then, for every task, explain every subtask to make chunks of time, making them easier to manage, but maybe more difficult to estimate the dedicated time. Once we have them defined, do a brief description of it and its subtasks, and assign time dependencies between tasks, because there are tasks that need to be completed before starting other ones.

4.1 Description of tasks

I have subdivided the tasks logically into 6 groups: Project management, Individual pre-development, Data warehouse, Portal and Documentation. They are defined in a chronological point from start to finish, but there are some that stay from the beginning, like the Documentation. This part of time order and dependencies, will be seen better in the Gantt Chart.

Project management

This iteration, helps me plan ahead of time the project. From zero, and with the help of the university teachers, we use a manual to learn how to manage projects from multiple points of view and areas of knowledge.

- **Contextualization and scope**

  The first task of the project management part is the definition of the scope of the project, important to define its limits and its context. 30 hours were needed for its completion.

- **Schedule**

  The second task, is defining tasks, a brief explanation, time and dependencies for each of them. 20 hours were dedicated to it.

- **Budget and Sustainability**

  After the tasks are defined, we need to estimate the costs of each task, resources and sustainability of our project. 15 hours were used.

- **Formatting document**

Due to my newness in LaTeXand the multiple table figures, I spent 5 hours to the crafting of the final document, making sure the margins were respected, and the document was visually appealing.

Individual pre-development

In this iteration, I research the market technologies that could be used for developing the project, test them and see the viability and document the information and results, for later explaining to the coworkers the investigation done.

- **Research state-of-the-art**

  Mainly research the internet for existing technologies, seeing alternatives and comparing different blogs and opinions. Thinking both back-end and front-end. 60 hours are dedicated to this job.

- **Testing technologies**

  After the research has been conducted, testing of the different technologies must be done to make sure it meets the requirements. The other part of this task is sharing the result to my supervisor, to get feedback and ensure he finds suitable what the technology gives us. Approximately 50 hours.

- **Documentation**

  An estimated 10 hours are dedicated to the documentation of the pre-development phase, that chronologically happens while the research and the testing are done. Giving us a theoretical and a practical view of it.

Team pre-development

When the team comes together, we get to meet each other and receive training about DCIM and company work methodology, it comprises a week (40 hours), and is on-site in Madrid. Some time is spent receiving teachings from multiple people in the company, and some other in documenting the knowledge received.

- **Training**

  Teachings about the technical knowledge are the main part of the team pre-development, but it also comes with talks from the CEO, marketing and HR team. 30 hours dedicated to it.

- **Documentation**

  A collaborative document will be made by us, to summarize the teachings received. 10 hours estimated.

Data warehouse

The first part of the development is the back-end. Looking for the data sources and building the back-end, giving us a solid back for building the front-end on.

- **Data warehouse finding**

  First part of it is the data source finding, as we work with many DCIM software products, it will be a difficult part, looking through different technologies. So we will dedicate a part to Training ourselves to work with SOAP, Rest and SQL queries to find data. And once we are done with the training, the find and check through trial and error will be the hardest part.

- **Training**

  An estimated 20 hours are dedicated to the training and research.

- **Find and check**

  80 hours will be dedicated to finding and checking data. It is a major part of our development, because as it is major to find data, we then need to check if the data that the software products shows the same data we see in the data source.

- **Data warehouse building**

  The second part of the data warehouse development task is the building of the actual DW, finding a DW architecture that better structures our data and makes it more efficient to read, write and scale. 100 hours estimated.

Portal

The second part of the development is the front-end development, known also as the Portal, a website to see the many data visualization modules and tabs.

- **Research**

  Research in front-end has to be done to ensure that the technology chosen for the website is the one that best fits the data. From the architecture to the data visualization modules. Thinking both client and server-side. 30 hours.

- **Data visualization**

  Data visualization development is crucial for the project, so 90 hours will be dedicated to it, checking the information visualization that explains the data in the way we want to do it.

- **Website development**

  The last 30 hours will be dedicated to the website part, as myself is not as good as the teammates in the web development, I will dedicate the most part of my time in the Data visualization field. Though some time will be spent here, it is difficult to estimate.

Documentation

It is done through all the duration of the project. Making sure no part misses information or an explanation. There is the final document part, with its imagery, tables, references, that occupies the majority of the documentation task. The other part comprised the preparation of the Oral presentation.

- **Writing final document**

  An estimated 80 hours are dedicated to it. It is being done with LaTeX, and citations, figures and tables have to be prepared in order.

- **Oral presentation**

  A total of 15 hours are used to the oral presentation of the bachelor thesis.

Below, a table to summarize the different tasks, its estimated completion time and dependencies between them, the bold numbers are the sum of the time of the subtasks that comprise the group.

Id	**Task**	**Time (h)**	**Dependencies**
T1	**Project management**	**70**	
T1.1	Contextualization and scope	30	
T1.2	Schedule	20	T1.1
T1.3	Budget and Sustainability	15	T1.2
T1.4	Formatting document	5	T1.3
T2	**Individual pre-development**	**120**	
T2.1	Research state-of-the-art	60	
T2.2	Testing technologies	50	T2.1
T2.3	Documentation	10	T2.1, T2.2
T3	**Team pre-development**	**40**	
T3.1	Training	30	
T3.2	Documentation	10	T3.1
T4	**Data warehouse**	**200**	
T4.1	Data source finding	100	
T4.1.1	Training	20	
T4.1.2	Find and check	80	T4.1.1
T4.2	Data warehouse building	100	T4.1
T5	**Portal**	**150**	
T5.1	Research	30	
T5.2	Data visualization	90	T5.1
T5.3	Website development	30	T5.1
T6	**Documentation**	**95**	
T6.1	Writing final document	80	
T6.2	Oral presentation	15	T6.1
	Total	**675**	

Table 1: Task definition, time and dependencies

4.2 Gantt diagram

Below, the Gantt diagram of the tasks.

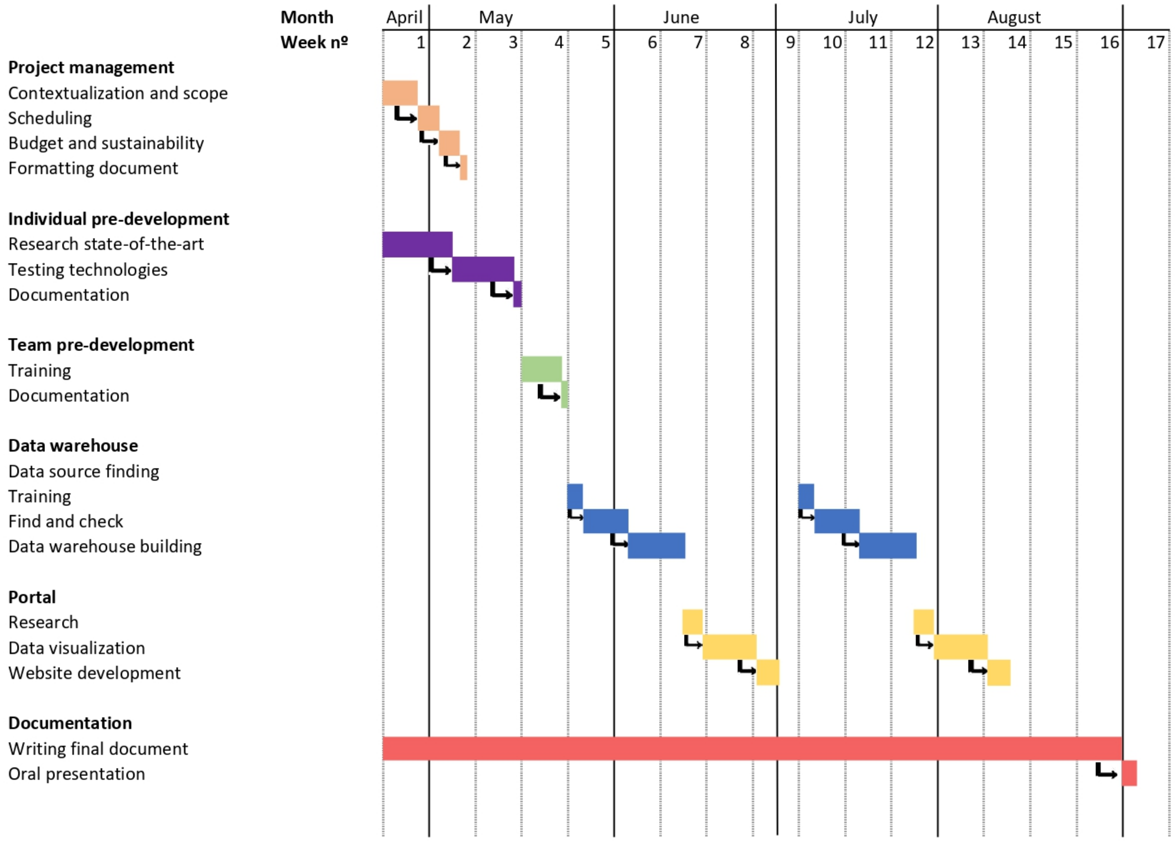
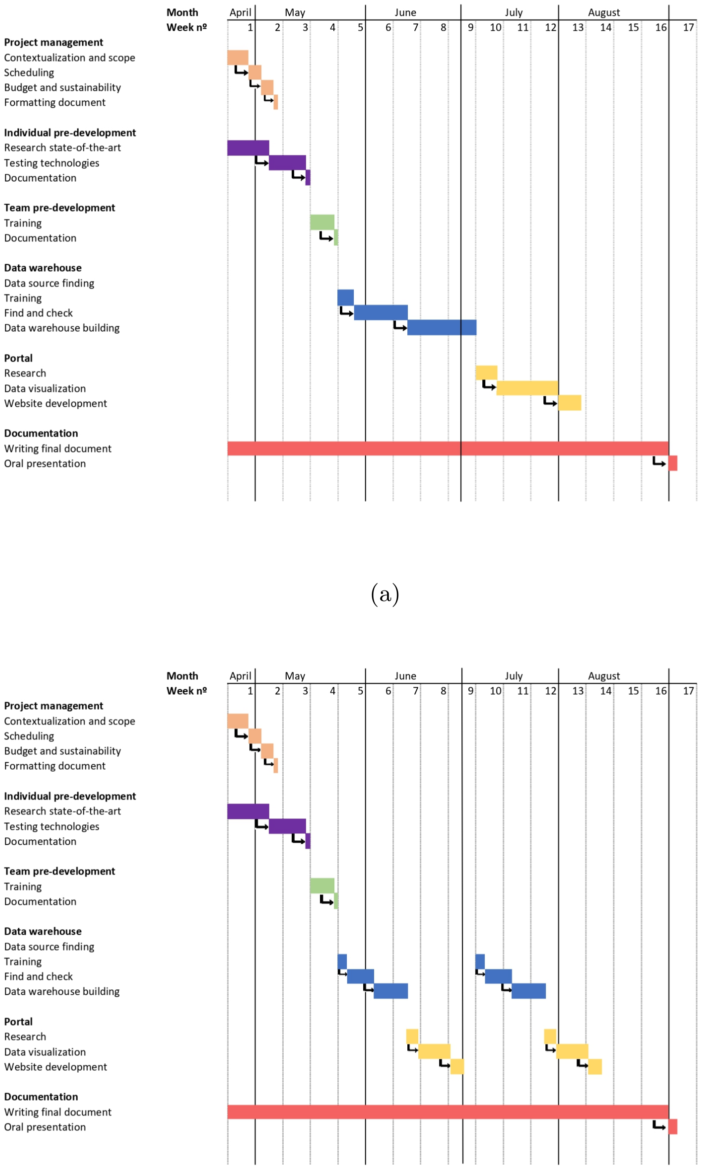


Figure 8: Final approach Gantt diagram: the final weekly scheduling diagram



(a)

(b)

Figure 9: (a) Is the initial approach, where we thought we would do all the data research, design and storage first, and then the data visualization and web design. However, the strategy was changed when we encountered difficulties in data finding.(b) Iterations on the cycle of data search, thereon data visualize, would be done to make data searches more efficient and more dependent on the data visualization we want to make.

4.3 Resources

Human Resources

The main human resources are the director, Antonio Suárez, the tutor on the project Pere-Pau Vázquez, and finally the innovation team at Bjumper, consisting of Achraf, Ericka, and me. It is substantial to understand that this project management is only done for my tasks in the project, that can be divided into two roles, the Project Manager and the Junior Developer. In a bigger company, two or more people might be dedicated to the tasks that I am developing.

The project manager will be in charge of conceiving the project, managing it, training the junior developers, part-researching for designing the solution and finally analyzing the product. This job could also be divided into solution architect, analyst and senior developer, but I have simplified it for the sake of making it comprehensive. The Junior Developer will be in charge of researching, testing, developing and documenting all the process. Below, the summary table.

Hardware

I will count on a **Thinkpad t470s** laptop, my personal computer, a second **Dell display**, for working better. **Servers** from the company, where we store databases, applications and where the team and I will run some tests.

Software

- **Overleaf:** To write documentation, the managament final document as the final thesis document.
- **Microsoft 365:** As a company communication, storing and sharing application.
- **Odoo:** As a company task time tracking web-based application.
- **Github:** To upload software developed in the innovation team.
- **Soap UI and Postman:** To explore DCIM applications Rest and Soap APIs.
- **Microsoft SQL and PostgreSQL:** To explore DCIM applications databases.
- **iTRACS DCIM and Schneider DCE/DCO:** For checking if the data found is equal to the data shown in the software.
- **IDE:** To be defined, maybe Visual Studio.

Id	Task	P.Manager (h)	J.Developer (h)	Total (h)
T1	**Project management**	**70**		**70**
T1.1	Contextualization and scope	30		30
T1.2	Scheduling	20		20
T1.3	Budget and Sustainability	15		15
T1.4	Formatting document	5		5
T2	**Individual pre-development**	**30**	**90**	**120**
T2.1	Research state-of-the-art	30	30	60
T2.2	Testing technologies		50	50
T2.3	Documentation		10	10
T3	**Team pre-development**	**30**	**10**	40
T3.1	Training	30		30
T3.2	Documentation		10	10
T4	**Data warehouse**	**40**	**160**	200
T4.1	Data source finding	20	80	100
T4.1.1	Training	20		20
T4.1.2	Find and check		80	80
T4.2	Data warehouse building	20	80	100
T5	**Portal**	**10**	**140**	150
T5.1	Research	10	20	30
T5.2	Data visualization		90	90
T5.3	Website development		30	30
T6	**Documentation**	**95**		95
T6.1	Writing final document	80		80
T6.2	Oral presentation	15		15
	Total CPA	**275**	**400**	**675**

Table 2: Estimation of project costs per activity (depending on role dedication to tasks)

5 Development

5.1 Pre-development

As mentioned in the Justification part, one of the DCIMs that does the best dashboard types of data visualizations is SunbirdDCIM, using the technology Grafana. A software where you select the source of data, and you can select or filter it and choose the type of chart or data visualization you want to represent that data. For this pre-development part, we decided it was better to work with a well-known and used data visualization, that is the occupation of a row, rack by rack by stacked columns, as a stacked chart.

5.1.1 Environment preparation

I worked with a Windows 2012 server that was at disposal from the company, the drawback was that it wasn't based in GNU, as I worked with Open Source based applications. Then, I installed the last stable version of Grafana.

5.1.2 Rack distribution data

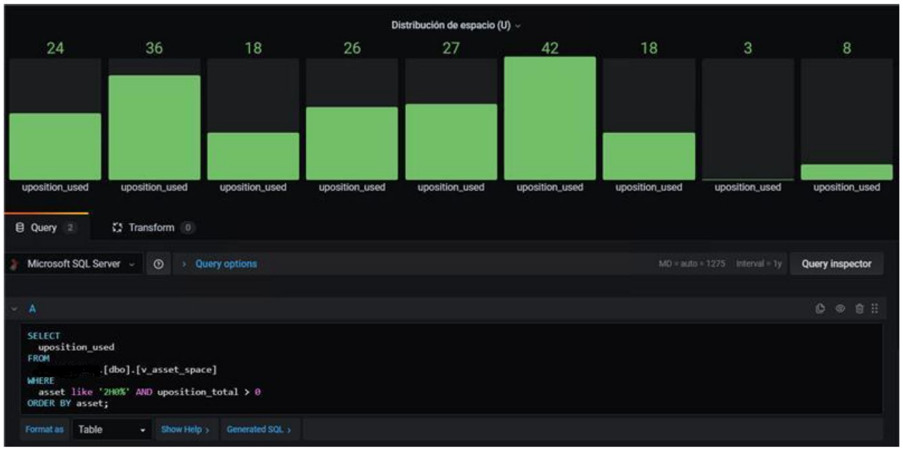
To select a Data Source in Grafana we have to go to the IP of where the DB is located and log-in with username and password. If we have explored the DB before, we know what SQL query gives us the data we need. We figured that doing the following query profiting from a SQL view "v_asset_space" we obtained the space of the asset, and if we search all the devices named as 2H0_, we are going to get all elements from the row 2H of the floor plan.

```
SELECT * FROM   #.[v_asset_space] where asset like '2H0%'
```

And it gives us back a table with assets from row 2H and all its columns.

5.1.3 Bar Gauge

If we investigate the chart examples in Grafana, we cannot find stacked bar charts, the chart that most resembles it is the Bar Gauge, where for every row draws a bar of its value.



```
SELECT
  uposition_used
FROM
[redacted].[dbo].[v_asset_space]
WHERE
  asset like '2H0%' AND uposition_total > 0
ORDER BY asset;
```

Format as: Table

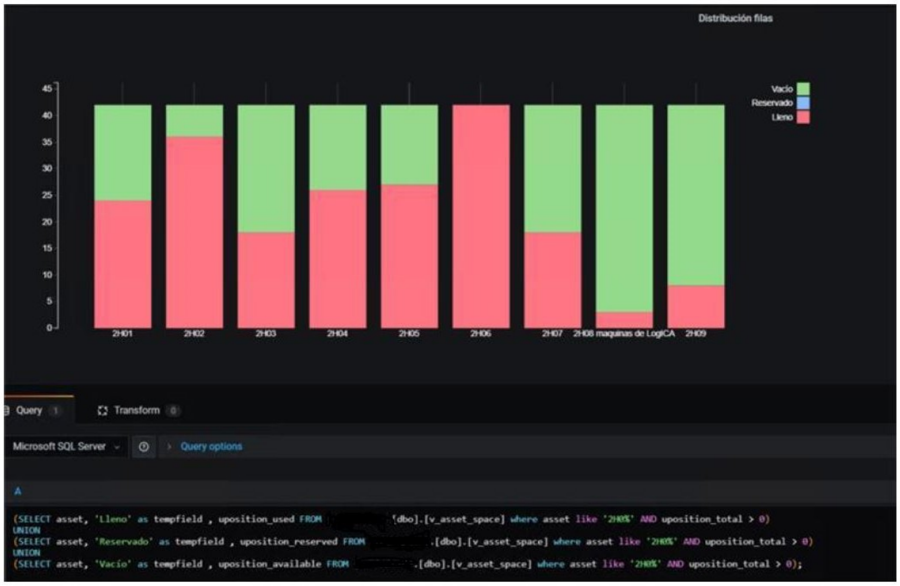
Figure 10: Bar gauge of rows occupancy distribution by row

Another limitation is that it selects the max value as the limit for every bar being the 42 U value, in figure, the max value, where there may exist racks with 30U that are full but in the bar gauge we can interpret as not full. In addition, there may exist racks of 45 U in the other data center rows, giving us not coherence between them.

One last disadvantage is that we cannot label rows properly by asset name, because it can take only the value column. For all these reasons, we had to search for ways to make Grafana visualizations more customizable, and we found the Grafana plugins, made by the community for doing custom charts or to receive data from custom data sources.

5.1.4 grouped-bar-chart

The plugin we chose for the stacked bar chart is named grouped-bar chart, a plugin that needs 3 values (Label, Type and Value). The label is according to the asset name, and it is placed under the bar, and will paint a bar with the height of Value, and the color assigned to the type variable. In the plugin configuration, we can change the color.



```
(SELECT asset, 'Lleno' as tempfield , uposition_used FROM [redacted] [dbo].[v_asset_space] where asset like '2H0%' AND uposition_total > 0)
UNION
(SELECT asset, 'Reservado' as tempfield , uposition_reserved FROM [redacted].[dbo].[v_asset_space] where asset like '2H0%' AND uposition_total > 0)
UNION
(SELECT asset, 'Vacío' as tempfield , uposition_available FROM [redacted].[dbo].[v_asset_space] where asset like '2H0%' AND uposition_total > 0);
```

Figure 11: Grouped-bar-chart racks occupancy distribution by row

The first limitation is the format that the plugin accepts, so that I have to do a triple query for every plugin and 2 unions with temporal fields to make a new table that meets the requirements. A solution to this would be the building of a data warehouse, putting the source data in the correct format and storing it in a table ready for Grafana to read it.

The second limitation is that the order of appearance of data in the bar is by the Type value in alphabetic order. So if we want to make the logic stacked bar chart that is the free above the occupied, we have to rename Ocupado to Lleno and Libre to Vacio to make it language coherent.

5.1.5 Conclusions

The main advantages of Grafana are the easiness, you can select a data source and a template and you will have a data visualization in minutes. But on the other hand, we saw a lot of limitations on the design, and because of these limitations, we chose to explore more customizable data visualization solutions. But before starting to customize any visualization, we have to ask ourselves why do we need this data visualization.

5.2 Task abstraction

To answer the question above, we have to go back to task abstraction 6, and see from the beginning why are we doing these visualizations. The main action is the Presentation action, centered in the decision-making, planning and forecasting. But it might sometimes be the Discover action, where we want to find unknown knowledge or validate a hypothesis. Or Search or Query for a middle user or Low user.

In this development part exist two major data-visualizations, which every one takes two weeks to learn the data vis language, search for data, send this data to the local database, and finally put them in the back-end of the web for the front-end to use and combine the data visualization language with the data. This is developed in the Agile methodology, so every data visualization is done through multiple iterations, improving in every cycle. In the following parts we are going to see the evolution of every part and finally see the results and validate them with the requirements given below:

1. It uses a defined task abstraction action.
2. It presents the data in a clear and easy way through usability tests.
3. If presented with interactivity, smoothness and usefulness of interactivity.

5.3 Data pipeline

Grafana gave us an architecture, really useful to use, just selecting the source and querying the data you could have a data visualization in minutes. But if we wanted to have more customization, we wanted to explore for other solutions fully sustainable on Open Source.

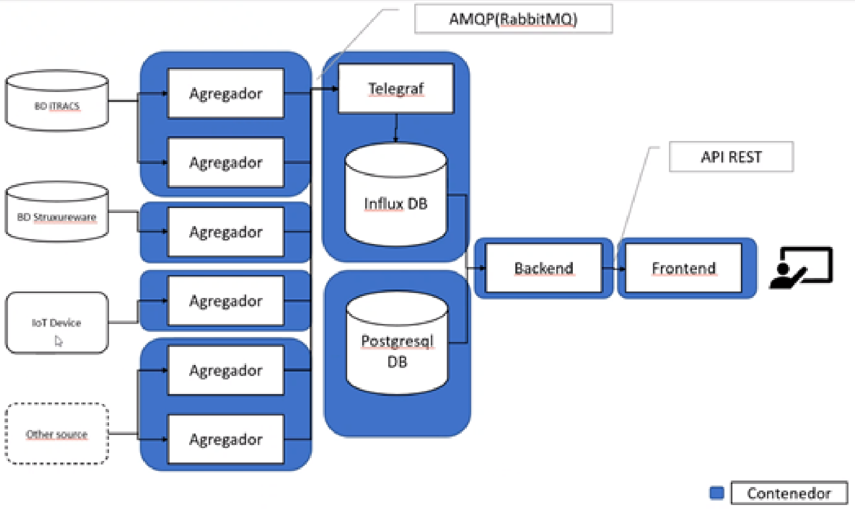


Figure 12: Architecture of our solution, in blue there are the different Docker containers and different aggregators for different data sources, the rectangles are data managing scripts normally in Node.js to connect different languages together. And the cylinders are the DB.

As we see in figure 12, the architecture designed is divided in 3 parts: Data aggregator, server and client. The data aggregator is dedicated to find the data in the sources and connect with a specified communication protocol, and then sending to a AMQP message broker. The server consumes the messages from the broker and stores them in a PostgreSQL database or InfluxDB, depending on the dataset availability. It also builds a back-end server developed in Node.js with a REST API. The clients uses the data visualization language and Node.js to read data from the REST API and visualize it.

5.4 Sprint 1: Vega

The Sprint 1

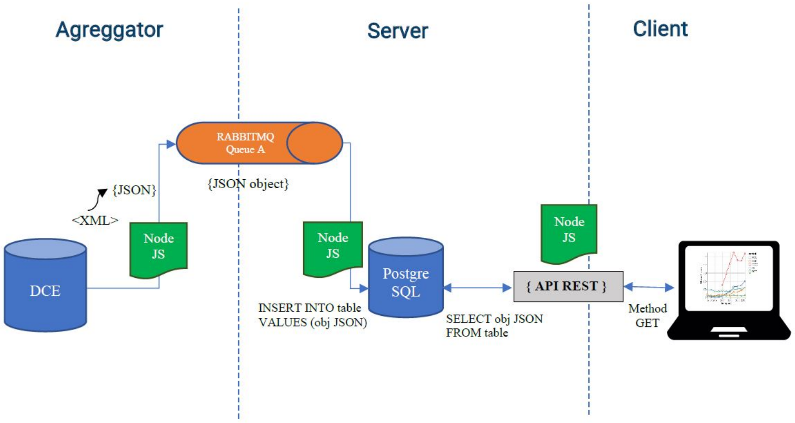


Figure 13: Architecture of sprint 1, we connect to with a SOAP protocol to get the data and send them to a AMQP queue, then read it and store it in a PostgreSQL DB and post them in an REST API. Then in the front-end visualize the data with an Angular front-end with Vega-lite components.

If we look into the chart, we can see that the data is getting bigger every day, because we sample data on a daily basis. So in the future, the data will occupy more horizontal space, a way to improve this solution is to add interactivity with a slider on time. If we had more time, we would definitely have tried this other iteration in the cycle to improve the data visualization.

5.5 Sprint 2: Vega-lite

The Sprint 2 is about doing a Discover/Query action type of task. With doing a top10 most occupied racks, we are presenting data in a way that we can Discover the racks that are most occupied, maybe to generate a hypothesis. And also a Query action when we are selecting data based on scale, either Site (location) where data center(s) are or room of the data center.

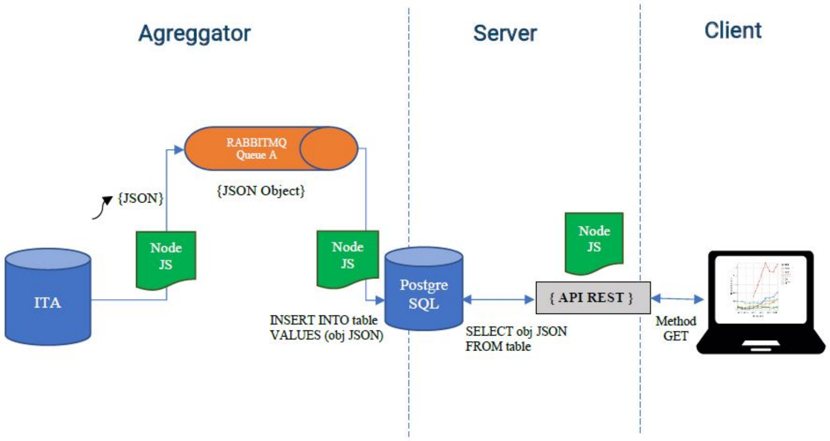


Figure 14: Architecture of sprint 2, using the same data pipeline as sprint 1, but the aggregator changes to reading from a REST API.

We are working with the same data pipeline as before, but instead of reading from a SOAP communication, we read with multiple REST API requests using Node.js.

6 Results

The results from Sprint 1 and Sprint 2 are the following, explained in the caption of the Figures 15, 16, 17 and 18 below and discussed in the next section called Validation.

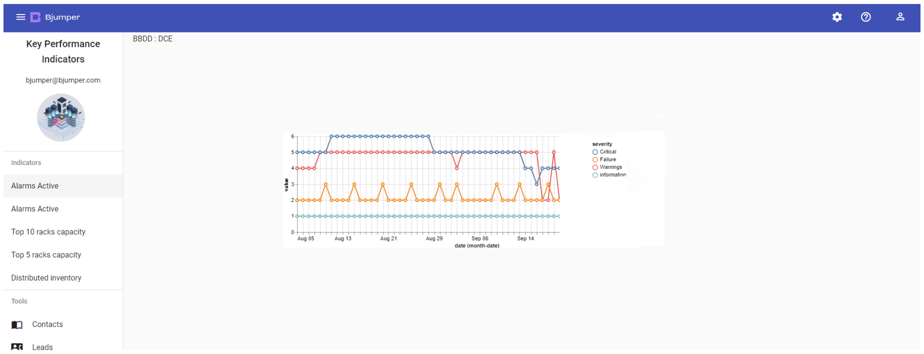
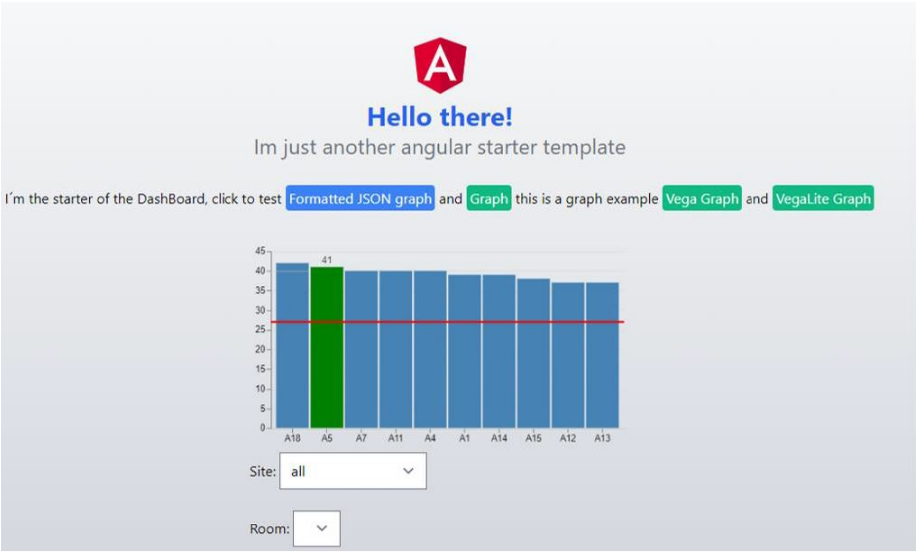


Figure 15: Result of sprint 1: Number of active alarms by type of alarm, and sampled on a daily basis.

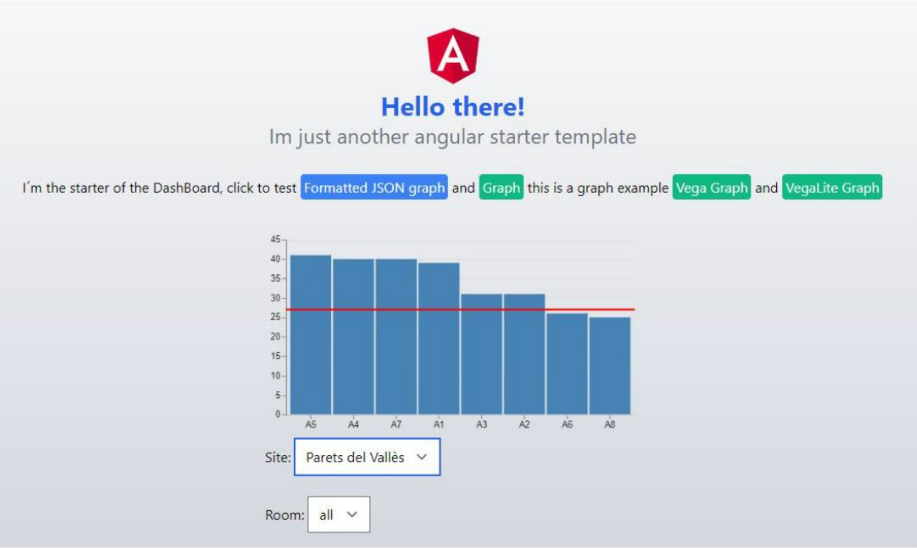


Hello there!

Im just another angular starter template

I´m the starter of the DashBoard, click to test Formatted JSON graph and Graph this is a graph example Vega Graph and VegaLite Graph

Figure 16: Result of sprint 2: Top 10 total racks from all the data centers from the business, when you hover on a bar it will pop green, you can see the value of occupied Us over it, and the green line marks the total average


Hello there!

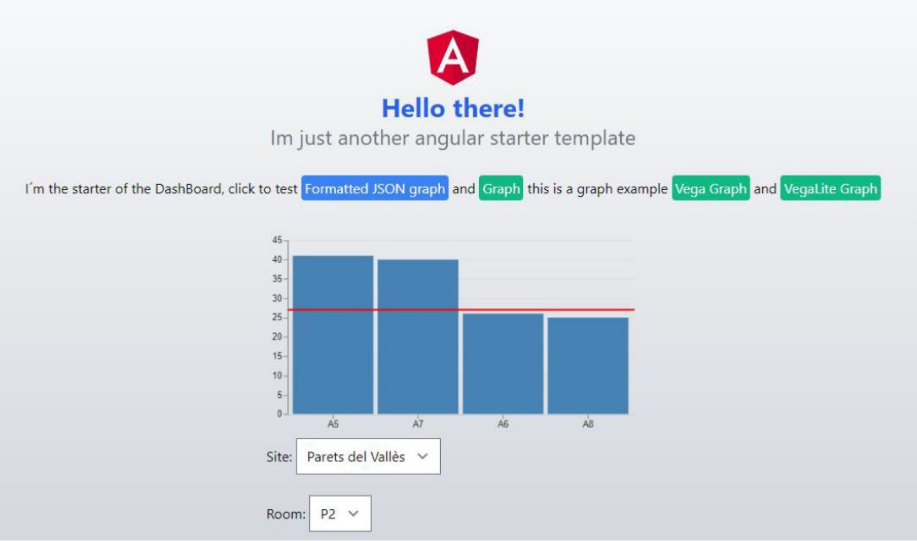
Im just another angular starter template

I´m the starter of the DashBoard, click to test Formatted JSON graph and Graph this is a graph example Vega Graph and VegaLite Graph

Site: Parets del Vallès

Room: all

Figure 17: Result of sprint 2: Interactivity with data, selection at location scale, to get the top 10 most occupied racks, and the red line marking the total average


Hello there!

Im just another angular starter template

I´m the starter of the DashBoard, click to test Formatted JSON graph and Graph this is a graph example Vega Graph and VegaLite Graph

Site: Parets del Vallès

Room: P2

Figure 18: Result of sprint 2: Interactivity with data, selection at room scale, to get the top 10 most occupied racks, and the red line marking the total average

6.1 Validation

To validate the results given above, the requirements that we set in the task abstraction section were the following:

1. It uses a defined task abstraction action.
2. It presents the data in a clear and easy way through usability tests.
3. If presented with interactivity, smoothness and usefulness of interactivity.

Both sprint 1 and sprint 2 data visualizations meet the requirements listed above. The first requirement is met with task defined actions, the tasks are done correctly and they both have made different usability tests from designers to users. The first one could have been tested with interactivity to make the time manageable, but we did not have resource time.

And the third requirement, only in sprint 2, because it has been developed with Vega and interactivity through querying the data, accomplished the smoothness and usefulness requirements of the interactivity. Finally, when we validate the data, we can improve the design and effectiveness of the solution through another iteration in the developing cycle. We can ask for more knowledge and new ideas to make the data visualization more visually appealing and clear. The cycle could go a long way, but given a limited time, the results were impressive. Following, we are going to analyze the entire solution given the scope defined.

7 Conclusions

In this thesis, we have designed new ways to visualize DCIM data by proposing a solution for better task-based data visualizations that facilitate the decision making. From researching the technologies, to testing and building new data visualizations, we have created custom data pipelines for different data visualizations, and different front-end technologies for different charts.

Firstly, in the DCIM monitoring data visualization research process, we found that the present different solutions that are doing dashboard type data visualization are failing in presenting them in a clear and understandable way. However, Grafana was the option which met most of our desired requirements, so was the chosen one to be tested. It exceeded our expectations in the ease and low time needed to build a data visualization, but disappointed in customization. So we tried to look for more customizable visualizations, building our own data architecture and website, front and back end solution.

Then, we defined actions that would determine different tasks for the data visualization. Based on those tasks, two temporal sprints were set. These contained several steps including its own researching, learning languages, data finding, managing and visualizing.

The following objectives have been accomplished in the following way:

1. **Research current DCIM monitoring technologies**

   First, research is crucial in the design of our technology, so searching for the current market solutions and alternatives is a must. After extensive research on existing DCIM monitoring technologies, Grafana was found to be one of the best ones mainly because of the ease of use but has many limitations in the design so more customizable solutions had to be explored.

2. **Define achievable tasks**

   Defining achievable tasks is primordial before searching the data. Before creating a data visualization, we need to define what we need that data for. With this step, we set our goal clearly, and it will define the rest of the process. They were defined through task abstraction method, and validated accordingly.

3. **Find data sources**

   We have to find the information we want to display in order to make our application functional. It was a more difficult task than expected, but we managed to get the data we wanted using different communication channels.

4. **Build data warehouse**

   A data warehouse that integrates many data sources, has to be designed in a well-thought architecture, making it easy maintainable and scalable. We built a functional data warehouse that is fast and easy to scale.

5. **Develop a data visualization website**

   This concerns the end user interface, the one which has the most importance. The data visualizations we did met the requirements defined. Presenting and representing the information in a clear and useful view.

6. **Make usability tests**

   During the multiple stages of this project, many usability tests have been made according to the Agile development methodology. Made for different types of users, from designer to low-skilled or high-skilled users.

Throughout the project we encountered some obstacles such as the lack of time to deliver a better data visualization or the underestimation time of tasks. Specifically, data finding, learning Angular, the front-end and learning Vega-lite data visualization language were the most unexpected time consuming activities.

To sum up, this study demonstrates that another way of doing DCIM visualizations is possible, meeting the requirements selected in the beginning of the project and doing so in a task-based and clear way. More iterations of the development cycle and usability tests should be made to fully obtain the final data visualization, but a great solution has been found. The results cast a new light on DCIM data visualization systems, and have a great recipe, mixing new programmers with DCIM experts and final users.

I believe the future of this project is bright, so I will continue to work on it because it is an interesting field where computers and design meet. Also, I find fascinating how humans and computers work together in order to accomplish decision-making tasks. There are many ways on how to try to approach the problem in the future but with the help of machine learning we could forecast the need for purchase of new facilities in the data center in the future, and visualize it with a trend graph, for the decision-maker to interpret.

Some of the best data visualizations have yet to come, made from people who are learning or will learn about them soon, because they will grow from seeds that have not yet been planted. I believe that the future is full of hope for the data visualization field.

8 Acknowledgements

This project has been possible thanks to the opportunity Bjumper and Antonio Suárez gave me to work in its innovation team.

Thanks to the help of my tutor Dr. Pere-Pau Vázquez for guiding me during all the project.

To my family, girlfriend, friends and co-students to helping and company throughout all the process.

Thanks to you.

9 Appendix 1: Budget

9.1 Identification of costs

First, if we want to quantify the cost of this project, we need to identify the different costs that intervene. The major cost of all is the human resource cost, that is defined in the costs per activity (table 3).

In order to get there, we have to define the costs of contracting a Proect Manager and a Researh developer, two different roles that are essential in this plan. Using Glassdoor, a platform for knowing the estimated salary of certain jobs, I have found that a Project Manager earns a net salary of 23 €/h and a Junior Developer 11 €/h. As we can see in the table 2. These values correspond to the net salary, so if we want to compute the total cost of the project, which the company is paying, we have to add the deductions and taxes that are applied. Social Security adds up to 33% of that quantity, so we will have to multiply the net pay to 1.33 to estimate the total cost. Below, you can see the estimated gross pay.

Role	Net pay (€/h)	Gross pay (€/h)
Project Manager	23	30.59
Junior Developer	11	14.63

Table 3: Cost per hour and role

Once we have the cost per hour of every role we have to assign tasks to different roles or different people, so that the cost will be different for every task not only because of the duration but also because of the implication of different roles. We repeat the table 1, but now with the assigning tasks to the different roles. Two roles have been designed for simplifying this case, where maybe some tasks like the Final document writing and Oral presentation have been assigned to the Project Manager, that I believe is closer than the Junior Developer of analyzing the end result and presenting it.

Project Manager will be in charge of the Project management part, and some parts in research and training for choosing a path for the Junior developer.

Junior Developer will be in charge of most part of the work, that is developing the actual product, also involved in a lot of research and testing.

To know these costs with clarity, we will have to see if the estimated times will deviate a lot from the actual hours dedicated to each task. See Management control section 9.2, for more information about deviations.

Below is the table with the hours dedicated to every activity, and the total cost of it. The cost of an activity comes from the sum of the Project Manager activity hours multiplied by its hour gross salary (30.59 €/h) and the Junior developer activity hours multiplied by its gross salary (14.63 €/h).

Id	Task	P.Manager(€)	J.Developer(€)	Total (€)
T1	**Project management**	**2141.3**		**2141.3**
T1.1	Contextualization and scope	917.7		917.7
T1.2	Scheduling	611.8		611.8
T1.3	Budget and Sustainability	458.85		458.85
T1.4	Formatting document	152.95		152.95
T2	**Individual pre-development**	**917.7**	**1316.17**	**2234.4**
T2.1	Research state-of-the-art	917.7	438.9	1356.6
T2.2	Testing technologies		731.5	731.5
T2.3	Documentation		146.3	146.3
T3	**Team pre-development**	**917.7**	**1064**	1064
T3.1	Training	917.7		917.7
T3.2	Documentation		146.3	146.3
T4	**Data warehouse**	**1223.6**	**2340.8**	3564.4
T4.1	Data source finding	611.8	1170.4	1782.2
T4.1.1	Training	611.8		611.8
T4.1.2	Find and check		1170.4	1170.4
T4.2	Data warehouse building	611.8	1170.4	1782.2
T5	**Portal**	**305.9**	**2048**	2354.1
T5.1	Research	305.9	292.6	598.5
T5.2	Data visualization		1316.7	1316.7
T5.3	Website development		438.9	438.9
T6	**Documentation**	**2906.05**		2906.05
T6.1	Writing final document	2447.2		2447.2
T6.2	Oral presentation	458.85		458.85
	Total CPA	**8412.25**	**6572**	**14984.25**

Table 4: Estimation of project costs per activity (depending on role dedication to tasks)

To compute the generic costs, we need to know if it is material or immaterial, and will exclusively be used for this project or the useful life is longer. For example, the material resources, we have to calculate its amortization, that comes from the (cost of the asset - salvage value) / useful life. I have determined the useful life to be 5 years and the use I will do for this project is 6 months approximately.

For the software costs, the only software that is not open-source, is the Microsoft 365 SaaS, looking online to know the actual price, I have found that for a company under 300 employees, the price is 5 €a month [15].

For the space costs, it has been more difficult for me, hence I have done an estimation, on my actual invoice.

Lastly, The transport part is a bit difficult to estimate as I have been once in Madrid, but I do not know if I will have to go again. For the moment, it stays like this:

Type	Cost (€)	Additional description
Hardware	**79.5**	
Laptop	66.5	Thinkpad t470s. Purchase price: 665
Screen	12	Dell screen. Purchase price: 119.99
Software	**30**	
Overleaf		Open-source
Github		Open-source
Microsoft 365	30	5€/month per user
SoapUI		Open-source
Postman		Open-source
Space	**600**	
Electricity	360	60 euros for 6 months
Internet	240	40 euros for 6 months
Water	90	15 euros for 6 months
Rent	540	20m2 by 4,5€/m2 for 6 months
Transport	**224**	
On-site stay	150	30 euros a night 5 days
Train	74	Travel with Ouigo Barcelona - Madrid round trip
Total	**1563.5**	

Table 5: Estimated General costs

Lastly, it is important to add the lasts 2 tables final costs together, the **CPA** (Cost Per Activity) and the **GC** (Generic Costs). Giving us a total of **16547.75 €**. I have established a contingency of the 15% as seen on other sites. And have divided incidentals on the types of risks described in the Obstacles and Risks section (3.2). The error that would cost us more budget is the Design error, would suppose the most part of the Incidentals cost, so we need to be watchful in the first design steps, to avoid it as much as we can. Finally, we get a total estimated cost of the project of **19705.91 €**.

Type	Cost (€)	Additional description
CPA + GC	**16547.75**	
Contingency	2482.16	Contingency of 15%
Total + Contingency	**19029.91**	
Incidentals	676	
Design error	368	40 hours of project manager, risk: 40%
Implementation error	132	20 hours of developer, risk: 60%
Temporal	132	30 hours of developer, risk: 40%
Technical problems	44	20 hours of developer, risk: 20%
Total	**19705.91**	

Table 6: Estimated Total costs, contingency and incidentals

9.2 Management control

In this section, we are going to describe the ways in which we will supervise and control the budget during the duration of the project. The difference between the real cost and the estimated cost is one of the main indicators we will use, though understanding the deviation in the former estimation.

For example, to know the cost deviation:

$$Cost\,deviation = (Estimated\,cost - Real\,cost)\,x\,Real\,hours$$

And for the deviation of the consumption of resources, we need to select the difference of time, (human resources) and multiply it by the estimated cost of the activity.

$$Consumption\,deviation = (Estimated\,hour - Real\,hours)\,x\,Estimated\,cost$$

Normally knowing the Real cost can be a bit tricky, but one way in our company to know the real cost of the human resource is using the **Odoo** technology, that enables us to track times to certain activities, so a person may be working in multiple projects in one day but input each hour to the corresponding project.

In the best case, the deviation is positive, that means that we overestimated the total cost of the project, we will reallocate the surplus money to other parts of the project, other projects or savings.

However, it is not always like this, and most of the time we will have to use the funds reserved to what cause the incident to cover the expense. If the cause is associated with any of the risks planned, we will use those funds, if not, we are going to access the contingency funds. If the deviations exceed all contingency and risk money, we will need to adapt the project to the budget available.

It is important to know the total real cost of the project, because it will be sold, and if we want to recover the investment, the price of the product will be calculated using the real cost. But the intentions of the CEO and marketing team may be other ones, could be to position the company in a new field or to deliver a technology to faithful clients for no return in the near future, rather as a long time investment. Anyway, knowing the real cost is key to the business.

10 Appendix 2: Sustainability

Doing a sustainability report is key to understand how this project will impact in the environment we live. Many companies are obliged to do it, and here I will try to conduct a sustainability report using the methodology explained in the Sustainability Analysis for the Bachelor's thesis. For achieving this, we have to analyze every impact : **Environmental, Economic** and **Social**. And for every type of impact we have to do a report on 3 levels. **Project Put into Production (PPP)**: that includes the planning, development and implementation of the project, **Exploitation**: that comes after it is implemented and ends when it is dismantled, and **Risks**: that are in every step of the process.

10.1 Environmental impact

Regarding the PPP, have you thought about the environmental impact of your project? Have you considered minimizing this impact, by for example reusing resources?

The environmental impact during the PPP is hard to estimate, due to the hours of use of computers, servers and energy needed for our project to be researched and implemented. Our website will also be running on our reused servers, so at least we are going to be able to manage the optimum resources to consume the necessary, and also reuse servers, so we are not gonna need to buy them new and generate more.

Regarding the life expectancy, how is it solved the problem you are trying to solve? Does your solution provide any improvement in the environmental impact?

Our DCIM management solution, makes power, cooling, network and space analysis easier, consequently, we help reduce environmental footprint in data centers by knowing better your infrastructure. Increasing efficiency and avoiding bigger problems that would cost more resources. Knowing the data is key in taking a decision about where you can do better environmentally.

10.2 Economic impact

Regarding the PPP, have you estimated the impact that your project will have, including both human and material costs?

In the Budget section, you can find the estimated economic impact that this project has, human and material resources, also with a contingency margin and potential deviations. After the implementation we will be able to know for real the economic impact both in human and material costs.

Regarding the life expectancy, how is it solved the problem you are trying to solve? Does your solution provide any improvement economically?

With this product, the main importance is in the economic field, knowing where you lose money or where you could earn a little more, managing resources efficiently can make a difference on your data center. We are trying to make a change in the data visualization field, making it easier to visualize your data center situation. The data warehouse also gives us the possibility to scale up the data and make predictions with ML.

10.3 Social impact

Regarding the PPP, how do you think this project will enrich you personally?

I have a huge opportunity in this project to work in a team, deign and implement a technology from the beginning to the end. With this opportunity comes great responsibility, as to the success of the project. Also working for the first time in a company, learning its structure and methodology, it will give me a new point of view of the business world, and it keeps me motivated to work with a supervisor and a director, which I will enrich from.

Regarding the life expectancy, how is it solved the problem you are trying to solve? How do you think your solution will improve people's quality of life? Is there a real need of developing your solution?

It may be believed that our solution is not market necessary, but in reality, no one does what we are developing. The knowledge that our company has combined with the energy of new people and new ideas might make a great solution. Data centers information is poorly viewed everywhere, and we try to tackle the problem with new ideas. And it is important that our visualizations are understood by everyone, so the people are the ones who matter the most to the project.

References

[1] iTRACS-DCIM.

[2] EcoStruxure IT Advisor | Schneider Electric | EcoStruxure IT.

[3] dco | EcoStruxure IT.

[4] Data Center Expert - APC USA.

[5] Tamara Munzner. *Visualization analysis and design.* A.K. Peters visualization series. CRC Press, Taylor & Francis Group, CRC Press is an imprint of the Taylor & Francis Group, an informa business, Boca Raton, 2015.

[6] DCIM - Data Center Infrastructure Management Software System, Cable Management, Infrastructure Design & Optimization Companies - Sunbird DCIM.

[7] Martin McKee, May C I van Schalkwyk, and David Stuckler. The second information revolution: digitalization brings opportunities and concerns for public health. *European Journal of Public Health*, 29(Supplement_3):3–6, October 2019.

[8] Evolution of Data Centers [A Look at how Data Center's have evolved over the past 50 years], August 2018.

[9] Andy Kirk. *Data visualisation: a handbook for data driven design.* SAGE, Los Angeles London New Delhi Singapore Washington DC Melbourne, 2016.

[10] Colin Ware. *Information visualization: perception for design.* Interactive technologies. Morgan Kaufmann, Waltham, MA, third edition edition, 2013.

[11] Ralph Kimball and Margy Ross. *The data warehouse toolkit: the definitive guide to dimensional modeling.* John Wiley & Sons, Inc, Indianapolis, IN, third edition edition, 2013. OCLC: ocn840431951.

[12] Grafana: The open observability platform.

[13] Time Series Database (TSDB) Explained | InfluxDB.

[14] Zabbix :: The Enterprise-Class Open Source Network Monitoring Solution.

[15] Tim R. A Guide to Microsoft 365 and Office 365 License Types.