UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

FIB

Aktana Business Intelligence

Josep Maria Canela Domingo

End of Degree
Information Technologies

FIB representant: José Maria Barceló

22 - 06 – 2022

Index

Image Index

Table Index

1. EOD Abstract

1.1. English

This is my End of degree (EOD) from Informatic Engineering Grade in Barcelona's Informatics faculty (UPC).

This project was proposed by Aktana. Aktana is the company where I have been doing an internship and later, I got hired as a Delivery Engineer becoming part of the company's crew.

Aktana is a pharmaceutical company that using embedded artificial intelligence refined by real-time human insight, Aktana gives life sciences commercial teams the information they need to improve the customer experience.

In this project it is going to be developed a Business Intelligence (BI), which is a system to show metrics and graphics in order to allow the directors take better operation management decisions.

The objective of this project was by part of the Aktana's Operation team to expand and incorporating new technologies as main purpose enhancing the company's performance to become more competitive in the market as the company is growing and every time is more difficult to track the employees' reported hours.

With this project I want to achieve to free up many hours form the director as it will be automatized.

Having the project ended, we can conclude that the project was successfully developed but we can't still see the impact in the market as is too early to see that.

1.2. Català

Aquest és el meu Fi de Grau (EOD) del Grau d'Enginyeria Informàtica a la Facultat d'Informàtica (UPC) de Barcelona.

Aquest és un projecte proposat per Aktana. Aktana és l'empresa on he estat fent pràctiques i més tard em van contractar com a enginyer de lliurament passant a formar part de la tripulació de l'empresa.

Aktana és una empresa farmacèutica que utilitzant la intel·ligència artificial incrustada perfeccionada per la visió humana en temps real, Aktana ofereix als equips comercials de ciències de la vida la informació que necessiten per millorar l'experiència del client.

En aquest projecte es desenvoluparà una Business Intelligence (BI), que és un sistema per mostrar mètriques i gràfics per permetre als directors prendre millors decisions.

L'objectiu d'aquest projecte era per part de l'equip d'Operacions d'Aktana ampliar i incorporar noves tecnologies com a objectiu principal millorar el rendiment de l'empresa per ser més competitiva en el mercat ja que l'empresa creix i cada vegada és més difícil fer un seguiment de les hores reportades dels empleats. .

Amb aquest projecte vull aconseguir alliberar una gran quantitat d'hores del director ja que serà automatitzat.

Un cop finalitzat el projecte, podem concloure que el projecte es va desenvolupar amb èxit, però encara no podem veure l'impacte al mercat, ja que és massa aviat per veure-ho.

1.3. Español

Este es mi Fin de Grado (EOD) del Grado de Ingeniería Informática en la Facultad de Informática de Barcelona (UPC).

Este es un proyecto propuesto por Aktana. Aktana es la empresa en la que he estado haciendo una pasantía y luego me contrataron como Ingeniero de entrega y me convertí en parte del equipo de la empresa.

Aktana es una compañía farmacéutica que utiliza inteligencia artificial integrada refinada por conocimiento humano en tiempo real, Aktana brinda a los equipos comerciales de ciencias de la vida la información que necesitan para mejorar la experiencia del cliente.

En este proyecto se va a desarrollar un Business Intelligence (BI), que es un sistema para mostrar métricas y gráficos con el fin de permitir a los directores tomar mejores decisiones.

El objetivo de este proyecto fue por parte del equipo de Operaciones de Aktana expandirse e incorporar nuevas tecnologías como objetivo principal mejorar el desempeño de la empresa para ser más competitiva en el mercado ya que la empresa crece y cada vez es más difícil hacer un seguimiento de las horas reportadas de los empleados.

Con este proyecto quiero lograr liberar una gran cantidad de horas al director ya que estará automatizado.

Habiendo finalizado el proyecto, podemos concluir que el proyecto se desarrolló con éxito, pero aún no podemos ver el impacto en el mercado, ya que es demasiado pronto para verlo.

Thanks

I thank all the people who have made this project possible or have shown interest in knowing your status, making special mention:

To the Aktana's colleagues who helped me when I needed help

To my co-director Raul Jimenez Alfonso for the help provided and to my director Aurelia Pagano

To my classmates for supporting me along this path,
particularly those who have accompanied me in my specialty.

Finally, to my family for allowing me to study this career and get here.

2. Introduction and contextualization

2.1. Document structure

The document is structured by apartments and sub apartments, of so that each apartment counted different parts of the project while all the subparts of the same apartment have something in common. The apartments are the following:

Abstract: In this section is fa a tiny description of what which consists of this project, both in Spanish and in Catalan and English

Introduction and contextualization: Here are the reference to all that is going to be done before of the development of the project, during the management phase.

Scopes of the project: Here are listed the project objectives

Methodology: Here is explained the way, the project, is going to developed

Temporal Planification: This section described the way this project is It has materialized and the steps that have been followed for this purpose.

Economic management: Project's budget

Sustainability: Here, is described the degree of sustainability of the project.

Justification: Why I have chosen this project

Tools: The software programs I have used to develop the project

BI development: Implementation of the Business intelligence

Conclusions: Project observations

Bibliography: Source from which the information has been taken from to be able to develop the project.

Annex: In this final section will be attached the codes and images related to the project

2.2. Context

This project is my End of Degree (EOD) of the Computer Engineering Degree at the Faculty of Computer Science of Barcelona (Polytechnic University of Catalonia). It has been developed for the Aktana's Operation team, with the main purpose of incorporating new tools or technologies to improve the work done by the employees and enhance the decision-making process.

Image 1. Aktana

The Image 1 we see, is the Aktana's logo. Aktana is a company who sells artificial intelligence to the customers. They work for pharmaceuticals such as MSD, Pfizer, Almirall, etc. The pharmaceutical world has all the time been in a prodigious side but after the covid this changed as it was a large time without selling drugs to the people.

One of the greatest Aktana's benefit is that they make the medical reps' life easier as they are told where they have to go, when they should go (this includes even the best call minute) and what they have to tell the doctor to increase the chances the doctor buys his product.

The optimization of the management is every time more important in terms of being as competitive as possible in the market. For doing that the best way is to have the maximum knowledge as possible about how the employees destinate their time for making the company work, larger the knowledge larger the capacity of detecting where the company is not as much optimal as it could be. Knowing that information we can deliberate which changes have to be done in order to improve the company's productivity.

With this project I pretend to make possible to create a Business Intelligence which will generate metrics and graphics that will allow the managers take better decisions, making possible a better company performance.

The project will be carried out using databases of type relational about MySQL as it is the companies' database engine, and it works for our project as we are developing a relational database.

We will import the necessary data to transform by using KNIME which is an ETL[1] (Extract, Transform and Load) that we will be using to import the data coming from the API.

If it's possible I also want to create a web-app where the manager could change the employee's information whenever he wants.

The project is based on the work done by a Delivery Lead who has already built the basic ETL tables and libraries. These libraries only took the data from the internet without doing any type of processing or transformation[2].

The loaded data will be transformed in the database and be shown as metrics and graphs in the corresponded dashboards.

2.3. Problem Formulation.

The problem was proposed by the Aktana's operations team as we needed to track the employee's performance in the projects,they are assigned in for a better knowing at how the resources could be better destinated as the company keeps growing has to have this record and information for it to be stronger versus its competitors by adjusting the prices as optimal as possible.

We are going to define these concepts:

1.1.1 Delivery Lead: director of a team. He has the role to coordinate all the three members team. He has to take decisions and communicate with the costumer whenever its needed.

1.1.2 Delivery Consultant: Between Delivery engineer and Delivery Lead. He is in charge of taking the client requirements and making Aktana's product changes. He has to be coordinated with the engineer.

1.1.3 Deleivery Engineer: Engineer of the team. He is in charge of making the technical develop in each release.

1.1.4 Operation team: The group of people in charge of taking the hiring/firing employees decisions in the company

1.1.5 Director: We are going to refer to him, as the director of all the operation team and is the person who wanted this project to be done

1.1.6 Trouble shooting: Is a task where the employees report on while solving a non-client problem.

---

[1]ETL: is a type of data integration that refers to the three steps (extract, transform, load) used to mix data from multiple sources. Often used to build a data warehouse

[2]Data transformation is the process of converting data from one format to another format that is more usable by the target system or application. Includes multiple activities: you can 'transform' your data by filtering it according to certain rules and joining different fields to obtain a consolidated view

Recently the company has started losing clients for the non-adjusted price and the expensive pricing when something had to be modified for some reason, we developed in a certain way and the client wanted in another way or the client has changed something in their system, and we must readapt to it.

To avoid this, the operations team developed a system to tack the employee's hours. Tracking the employee's hours, we can know where they are destinating them to. For example, if they are destinating too many hours to trouble shooting task, we know there is some problem in a part of the system company. To have this knowledge we need to gather all the information in one site. At the moment, the data was being tracked by being inserted by the operation team director which was a very time-consuming task. The director was having to spend more than 4 hours a week for updating the employee's information in the system. As it is an important role in the company his time in more valuable, so it was having a higher money impact. Moreover, his is supposed to be played to take decisions and wasting his time on inserting data to an excel is clearly a waste of time and it reduces the time he will deliberate who to take the decisions.

To end with the problem formulation Aktana wants to have the entry records system independent from TimeCamp. The reason is because TimeCamp deletes entries from long time ago to save space and not having to pay for extra space.

2.4. Art status

Aktana is losing clients for making them pay an elevate price for their services. For this in the past some people had to leave the company.

With this project we want to achieve a better performance of the company. To make this we have to start for the fundamental part of the company, the employees. The employees are the main part to make the company work, if we increase the employee's productivity will be translated into making then the company more competitive and more attractive for the costumers.

Also, will free up lot of time from our director in the sense that now this task is being done by updating all the employees reports and information by hand in an excel. This obviously is very time consuming from an executive role who should be destinating his time in more important things.

There is already a system to track the employee's hours. It is done by using an ETL to get the data from the TimeCamp API and staging it into the database entries table without doing any transformation later. Now we were only getting the last updated data, without knowing any changes the employees have done during the weeks. The ETL had to be ran every week manually and the tables had to be queried every time the data wanted to be check. Finally, the data was loaded into Tableau, which is a business intelligence application (explained at x), and there the delivery leads and the directors were able to see the employees metrics and graphs.

By having this we don't even had the information linked to the Aktana's employees, these reports where just showing the reports of the TimeCamps users, which means if some

TimeCamp user was having two or more assigned roles we were not knowing in which role where the hours reported. Imagine, if some employee is delivery engineer in the morning and in the afternoon, he is working as data engineer, then we could not be able to distinguish the hours reported by him.

To make this possible Aktana had a google sheet containing all their employee's information. As you imagine this sheet had to be filled every week or time some employee was being hired, fired or had a role modification. This sheet was being filled by the operation director. As I said before this was tacking him so much time and the company was not being as efficient as they could.

Also, the company wanted to know the employee's compliance. The compliance is to have the records when the employees report the hours, during the week as they are told to do or not. To have this record, now was being done by taking weekly snapshots of the entries table at our database. Another time this task was being done by the operations team director making him lose more time.

Unlucky until this moment the company had not had another way to solve these troubles and that's why they wanted to develop this business intelligence project.

2.4.1. Business intelligence

Business Intelligence is called the set of strategies, applications, data, products, technologies and technical architecture, which are focused on the administration and creation of knowledge about the environment, through the analysis of existing data in an organization or company.

The term business intelligence refers to the use of data in a company to facilitate decision making. It covers the understanding of the current operation of the company, as well as the anticipation of future events, with the aim of offering knowledge to support business decisions.

Intelligence tools are based on the use of an intelligence information system that is formed with different data extracted from production, with information related to the company or its fields, and with economic data.

Using the tools and techniques ETL (from the English «Extract, transform & Load»), or ETC (equivalent in Spanish: «extraction, transformation and load»), the data is extracted from different sources, purified and prepared (homogenization of the data), and then load it into a data store.

The life or period of success of business intelligence software will depend solely on the success of its use for the benefit of the company; If this company can increase its financial-administrative level and its decisions improve the performance of the company, the business intelligence software will continue to be present for a long time, otherwise it will be replaced by another that provides better and more precise results.

The basic requirements that make it possible to optimally organize Business Intelligence starts with having data that duly reflects "all the company's processes" and, preferably, is gathered in the same database, this facilitates access and monitoring. Have technological tools to

independently create all the dashboards and KPIs, and finally, the sectoral knowledge to know what to analyse in a coherent and strategic way.

Finally, analytics tools make it possible to model query-based representations to create a balanced scorecard that serves as the basis for reporting.

This is the solution I am going to implement to solve all the problems mentioned in the problem formulation part.

3. Scope of the project

As in any project, it is important to define its scope, since the development time is limited, and it is necessary to define the objectives and requirements of the application. Next, the objectives, the non-functional requirements and the obstacles and risks of the project are defined.

3.1. Project's objective

The main objective of this work is the development and implementation of a tool to evaluate the users and the tasks related to them, leaving a record of all the modifications made. Next, the objective is divided into the following sub-objectives:

Once defined and explained, in the second apartment, the problem pretending to solve, we can proceed to analyse the several objectives.

Now we are going to list the project objectives:

1. Enhance the company's performance.

2. Reduce the Operation team time consumption in the process of taking decisions.

3. Develop all the workflows to be able to make all the amounts and the filtering of the data that the company's employees report in Timecamp [2].

4. Develop all the part of the databases which implies the creation of the tables and the triggers to fill the historical tables.

5. Make a web-app so that the executives put the changes made in a clearer way.

6. Make our record system independent of TimeCamp

3.2. Possible obstacles

Next, we will analyse the problems or the difficulties that we could find during the execution of the project. Proposing for each of them solutions that reduce their impact on the process of carrying out the project.

- Errors in the code

- One of the obstacles that we will surely come across are errors when it comes to program. Any small error, if it is not solved in time, can make us waste many hours to solve it. Therefore, it is important to perform tests often on time in order to detect these failures as soon as possible.

- One employee deletes the entire database schema

- Calendar:
  Knowing that the time for carrying out this project is limited, it is necessary to set a calendar with deadlines. In such a way that the project is carried out. Always carried out progressively and with due supervision, with the objective of that we do not deviate from the main objective, and we can reach the correct realization of the Project.

- Changes in the databases:
  Although it is not common, we must consider that our application must access the databases of students, subjects, universities with an agreement... In the case of that the data or the format of these were modified, changes would have to be made in the application in order to keep our application working.

- Changes to the necessary requirements:
  During the execution of the project, it may be the case that some of the ideas taken at first, they end up not being done or that some new one is included. Therefore, we must be attentive to any modification that we believe we must make or that the client tells us that he wants to incorporate

3.2.1. Requirements

Next, the non-functional requirements of the system are defined, that is, those requirements that do not speak directly of the operation of the system, however, they must be considered from the beginning for the development of the project:

1. Usability – It is important to make the front part as intuitive as possible since the users will not know how the whole project is built.

2. Efficiency – Regarding this point, the requirements are not very difficult to achieve since our ETL will be executed once a day when the data is already entered. If it takes less than 3 hours to be updated, it would be enough.

3. Reusability - To facilitate the integration of the system in other applications, it is interesting that the implemented techniques can be integrated in different projects.

4. Methodology

The development of this project is related to the investigation of the world of business and the management of company resources. It is very important to know what the main needs of a business are and to know how to manage these resources in order to optimize time.

Specifically, following the agile methodology, weekly meetings will be held with the project directors, in which the weekly objectives will be verified, and depending on the status, modifications will be made to the planning or new tasks will be established. To make.

The project tracking will be done by the director and coo director by doing periodical meetings every week.

Regarding validation, tests are carried out once each part of the block is finished in order to verify that the entire architecture and operation of said block is correct and follows the requirements that users want.

In a company there is a certain number of resources allocated to the several tasks the company has to perform and it is important to manage these as optimally as possible. To make it easier to manage, we will implement a Business Intelligence (BI) refers to the use of strategies and tools that serve to transform information into knowledge, with the aim of improving the decision-making process in a company.

To make it possible we will need a large amount of data that later we are going to transform in statistics and metrics for our understanding so the company business operation team will be able to take proper decisions.

To get this data is needed the employee hours reported, emails, in which task the employee has worked, etc. In our company the employees report the hours in a platform called TimeCamp and that's where we are going to import the data from. Also, we are going to import data from a google sheet.
These employees report the hours worked on a website. We, with this project, want to analyse these and show relevant data to executives. So that they can access the data, we also want to develop the front-end part, which will consist of a web-app where you can see the graphs and metrics generated with our BI.

Thanks to this, managers will be able to see how resources are being managed and what impact they have on the company and then they will be able to make decisions according to the metrics taken and thus optimize the resources of said company.

This project, the entire back-end part will be developed first, and the entire front-end part once back is finished. For this reason, we will divide the problem into two blocks, on the one hand, the implementation of the BI and on the other, the web app for the BI.

4.1. Back end

To start with the development of the back-end, we must first adapt the entire ETL library to have the data that we are going to process loaded and updated. In order to achieve that we must know what will be the information of the tables that we will want to use. Once we know it, we can generate all the workflow needed to feed our staging tables from our database. The database should have two schemas[3], the staging one and the Data Warehouse schema, both isolated between them, but this project doesn't have the necessary budget to fulfil these conditions and we are going to develop it in the same schema. Not having the schema spitted and isolated can impact if some employee deletes the database, then it will be all deleted. Furthermore, we can have security gaps as we let access to all the tables just having one password.

Once the information has been imported, we will have to design the entire Data Warehouse. To do it, we must make a design according to the requests that our Delivery Lead will make us, who is the one who knows how the final product should be.

A mandatory requirement for the design is that it must be extensible for the future.

4.2. Front end

As a second problem we have to represent the data in an interface which is the most understandable for the user. To do this we must develop some dashboard where we will be able to graphically display the information obtained through the transformations. Finally, we will implement a web-app to make employee changes there and not having to access directly to the database

This information should only be accessible to company executives since we want it to be private and reserved information for those in charge of managing the company's resources.

[3]schema: The database schema is a structure of a database described in a formal language supported by the database management system (DBMS).

4.3. Involved agents

The system to be developed is aimed at the company's executives, so we will need their cooperation to design our system according to their requests.

As a result of this work, several executives will be able to interact through the web-app, a fact that will allow them to increase the productivity of their employees.

4.4. Designer and developer

The designing and developing part will be made by myself, the person in charge of the project.

4.5. Project director and coo director

The project director is Aurelia Pagano. She is mainly going to supervise the project. The coo director is Raul Jimenez. He will show me the guidelines and how the final user wants to have the system developed.

5. Temporal Planification

In order to complete this final degree project on the estimated date, and to meet the objectives previously set, in this section a temporary planning of the project divided into tasks is performed.

The work begins on February 8, 2022 and is scheduled for completion on June 17, 2022. In total, the development of the project will take place over approximately 5 months and with an estimated duration of 422 hours.

The daily dedication will be 4 hours a day, from Monday to Friday Business Intelligence system development tasks will be performed. The weekends will be dedicated to documentation.

5.1. Description of tasks

This section details the tasks to be performed individually, but they are grouped into blocks to more easily distinguish the different phases of the project. Table 2 lists all tasks with the required duration, dependencies, and resources, and Table 1 shows the initial project planning.

PM - Project management

Project management is essential to plan, define and document the work to be done, in addition, it includes meetings for the validation and proposal of weekly objectives. It is estimated that overall, the management group.

PM.1 – Scope

Before starting work, it is necessary to limit the development, for this reason, initial time is spent defining what you want to achieve with the job, what is going to be developed, and what means will be needed. Prior to the scope, it is necessary to have researched on the subject, as it is necessary to take it into account before defining which techniques will be developed.

PM..2 - Planning

To meet the proposed objectives in defining the scope of the project, temporary planning is carried out, as well as resources and associated with each task. In addition, risks and obstacles are defined, and alternative tasks are proposed to solve them.

PM.3 - Budget

A budget will be made to quantify the cost of the project, for this, items will be made for each task considering the costs of staff and equipment, in addition, generic costs and contingency items will be quantified.

PM.4 - Sustainability Report

The report will analyze the environmental, economic and social impact of the project, in particular, the planning and development phases.

PM.5 - Meetings

As with any research project, it is necessary to hold frequent meetings to analyze the results obtained according to the weekly objectives, and to decide whether aspects of the planning should be changed. 1-hour weekly meetings with project directors are planned. In addition, follow-up meetings will be organized meetings with managers to determine the design of the system.

PM.6 - Documentation

An important part of the TFG is the final report, therefore, throughout the development of the project, the different phases must be documented. The documentation will be carried out in parallel with the rest of the project, in this way the sections in which work will be incorporated. The estimated duration is 60 hours.

PM.7 - Presentation

Finally, once the documentation is completed, it is necessary to prepare the submission for the court that will evaluate the TFG. Support material will be prepared for the presentation, as well as the script, and rehearsals will be conducted.

PW - Previous work

This section specifies the tasks to be performed before the development of the work, ie preparation and prior study tasks. Since this is a preparation phase, it can be done in parallel with most project management tasks.

PW.1 – Study how to develop Knime Workflows

As far as this point is concerned, the preparation of the work environment is quite simple. You must download the ETL Knime and import the base libraries where we will make our connections.
Once there it will be needed to learn how to link the certain nodes and which more libraries are needed to be imported.

PW.2 - Study of the different types of DWH

I have to study to know how is going to be built the data warehouse in the most optimal way. This will take some time as I don't have knowledge about Data Warehouses, and I will need to start from the beginning. Once designed I will create a UML diagram to show the how the Data Warehouse structure should be making a helpful for the implementation.

PW.3 – Study of Business Intelligence

It is needed to study how the generated data from the Data Warehouse will be linked to the Business intelligence in order to see the data from the DWH as metrics and in a graphical way by using dashboards. This task will take

PW.4 – Study how to develop the Aktana's web-app

In this last section of the previous work, it is necessary to research which would be the best tool and language to develop the web-application always according to the director's guidelines.

DD - Development of a BI and WebApp

This section will const of to phases:

- The first one will be the Data Warehouse development including there also the Knime Workflows design and implementation together with the validation of these.
- The second phase will consist in the Web-app development including there the design, implementation and validation.

Duration 200 hours.

DD.1. Design

This part has a large weight in the project since have to be designed the Knime workflows in the proper way together with designing a Data Warehouse. Later, in the second phase, will be needed to investigate how to implement the company web-app.
This will be happening in the many phases of the implementation, so we will dedicate an overall of 75.

Phase 1 Design of the Business Intelligence:

- Design Knime Workflows: Design in paper the final Knime Workflows to make it easier at the implementation time.

- Design DWH: I will generate a UML diagram with the final Data Warehouse schema with the relations between all the tables from the database. Also, will include all the fields and keys as it is a relational model.

- Design of the Business Intelligence dashboards: This is the final part of the first phase design part. It is needed to study how to design the necessary dashboards in the business intelligence.

Phase 2 Design of the web app:

- Design Web app: Schema of the Web-app.

DD.1.2 Implementation

The implementation of the product is to translate the previously made design. Code programming is tighter with the research the most extensive and important part of the project. It has a big weight in the project, so we will dedicate a large amount of time, about 120 hours in total.

Inside the implementation part we have to distinguish between the two phases of the project there are the following task to be done:

Phase 1 Implementation of the Business Intelligence:

- **Knime Workflows implementation:** This is the first step of the implementation phase. The previous designed Knime Workflows will be developed in the Knime tool in order to be capable to import data from the APIs to our database.

- **Create staging tables with the triggers:** here are needed to create the staging tables with the corresponding trigger to populate the historical staging table for later the procedures from the Data Warehouse will populate their tables becoming the staging tables the first step of the Data Warehouse data transformation.

- **Implementation of the Data Warehouse:** once the UML diagram is finished the Data Warehouse tables will be created. Once created, will be needed to be populated. This is the most difficult part of the project as will be needed to be created stored procedures in order to populate those.

- **Create Business Intelligence dashboards:** This is the last step of the Phase1 implementation part. Consists on creating the dashboards in the business intelligence for letting us see the representation graphically and statically of the generated data at the DWH.

Phase 2 Implementation of the web-app:

- **Implementation of the web-app:** This is the final stage of the implementation part project, it consists on developing a full web-app for from there changing the employee attributes, such as role, name, id, etc.

DD.1.3 Validation

Validation will be performed in parallel with implementation as objectives can be validated. An 8-hour dedication is expected.

Like in the before parts this one also consists of two phases:

Phase 1 Validation of the Business Intelligence:

- **Test Knime Workflows:** Here will test the Knime workflows by comparing the data from the API versus the one loaded in the database staging tables. With this step we will be testing if the triggers from the non-historical tables are getting called making with that a more consistent validation of the next task "Test staging tables with the triggers".

- **Test staging tables with the triggers:** This task has the purpose of testing if the stage tables are consistent. Is needed to check both table data and compare between them. It is needed to check if the data in the "valid_from" and "valid_until" is properly populated.

- **Test Data Warehouse:** This task has to be checked with the old dashboards from Tableau. Is needed to check if the current dashboards are showing the same data as the old ones.

- **Test the Business Intelligence dashboards:** There is not much validation in the dashboards from the business intelligence as they just show the data in a graphical way. To check if the data is being well queried its necessary to compare the old dashboard values with the values from our business intelligence, the newest one. Once this is testes, we can conclude with the phase 1 of the implementation.

Phase 2 Validation of the web-app:

- **Validation of the web-app:** This is the last task of the implementation part. Consist on verifying the web-app is working as supposed. By checking the data is being modified also in the database.

After ending with this tasks, the main work of the project is developed. We can move to the next step to refine our project making the most optimal as possible. We also want to have the project automatized.

CP Develop a complete product

This task will consist on refining the code and create the parent workflows to encapsulate all the procedures from the Data warehouse and the ones from the ETL. After all we will deploy the ETL to the Aktana's Data Science Server.
It is the last phase of the project, and the remaining time will be devoted to this point. I t is 8 hours per design and validation and about 80 hours in implementation.

CP.1 – Design and research

The design tasks will also be spitted into the two phases:

Phase 1 Final BI Design:

- **Design of the Knime parent workflows:** It is necessary to make a research and design for later implement the parent workflow. Will be also needed to research about how to deploy the parent workflow to the server for letting the server automatized. For doing that we will also need to investigate how to schedule a Date&Time execution periodically.

- **Design the procedures:** Is needed to research how to design and implement a parent procedure.

Phase 2 Final Web-app Design:

- Research how to refine the web-app code

CP.2 – Implementation

Consists on implementing the previous designed task.

The Implementation tasks will also be spitted into the two phases:

Phase 1 Final BI Implementation:

- **Implement Knime parent workflows and server deployment:** This is the final step to do in the Knime side. In this task we will implement the parent workflow and once it is finished, it will be deployed to the Aktana's server. When deploying will be needed to schedule an execution Date&time as we want to execute the workflow every Monday at 12 AM at Spain.

- **Implement the procedures:** Implementation of the parent workflow.

Phase 2 Final Web-app Implementation:

- Refine t the web-app code

CP.3 – Validation

The validation tasks will also be spitted into the two phases:

Phase 1 Final BI Validation:

- **Validate Knime parent workflows and server deployment:** Check the workflows are being properly ran weekly at 12 AM

- **Implement the procedures:** Check the DIMs and the facts are being well populated.

Phase 2 Final Web-app Validation:

- Validate Web-app

After finishing all this steps, we can conclude the project as finished.

The following table show how the task will be scheduled and the overall estimated number of hours:

Id.	Task	Time	Dependency	Resources	Roles
PM	**Project management**	**57 h**	-	-	-
PM.1	Scope	5h	-	Laptop	D
PM.2	Planification	2h	PM.1	Laptop	M, P
PM.3	Budget	1h	PM.1	Laptop	D
PM.4	Sustainability report	6h	PM.1	Laptop, DBeaver	P
PM.5	Meetings	20h	-	Laptop	D, M,
PM.6	Documentation	20h	PM.6	Laptop,	P
PM.7	Presentation	3h	-	DBeaver, Knime	P
				Laptop	P
PW	**Previous Work**	**100h**	-	-	-
PW.1	Study how to develop Knime Workflows	30h	PM.1	Laptop, Knime	M, P
PW.2	Study of the different types of DWH	40h	PM.1	Laptop, DBeaver	M, P
PW.3	PW.3 – Study of Business Intelligence	20h	PM.1	Laptop, DBeaver	M, P
PW.4	PW.4 – Study how to develop the Aktana's web-app	10h	PM.1	Laptop	M, P
DD	**DD - Development of a BI and WebApp**	**200h**	-	-	-
DD.1	Design	40h	TP.2	Laptop, DBeaver, Knime	P
Phase 1	Design Knime Workflows	10h	PW.1	Laptop, Knime	P
Phase 1	Design DWH	10h	PW.2	Laptop, DBeaver,	P
Phase 1	Design of the Business Intelligence dashboards	10h	PW.3	Laptop, DBeaver,	P
Phase 2	Design Web app	10h	PW.4	Laptop	P
DD.2	Implementation	120h	DD.1	Laptop, DBeaver, Knime	P
Phase 1	Knime Workflows implementation	20h	DD.1	Laptop, Knime	P

Phase 1	Create staging tables with the triggers	30h	DD.1	Laptop, DBeaver	P
Phase 1	Implementation of the Data Warehouse	50h	DD.1	Laptop, DBeaver	P
Phase 1	Create Business Intelligence dashboards	10h	DD.1	Laptop, DBeaver	P
Phase 2	Implementation of the web-app	10h	DD.1	Laptop	P
DD.3	Validation	40h	DD.2	Laptop, DBeaver, Knime	T
Phase 1	Test Knime Workflows	5h	DD.2	Laptop, DBeaver, Knime	P
Phase 1	Test staging tables with the triggers	5h	DD.2	Laptop, DBeaver	P
Phase 1	Test Data Warehouse	20h	DD.2	Laptop, DBeaver	M, P
Phase 1	Test the Business Intelligence dashboards	2h	DD.2	Laptop, DBeaver	M, P
Phase 2	Validation of the web-app	8h	DD.2	Laptop	M, P
CP	**Develop a complete product**	**65h**	-	-	P
CP.1	Design	20h	DD	Laptop, DBeaver, Knime	P
Phase 1	Design of the Knime parent workflows	5h	DD	Laptop, DBeaver, Knime	P
Phase 1	Design the procedures	10h	DD	Laptop, DBeaver	P
Phase 2	Research how to refine the web-app code	5h	DD	Laptop	P
CP.2	Implementation	30h	CP.1	Laptop, DBeaver, Knime	P
Phase 1	Implement Knime parent workflows and server deployment	10h	CP.1	Laptop, DBeaver, Knime	P
Phase 1	Implement the procedures	10h	CP.1	Laptop, DBeaver	P

Phase 2	Refine the web-app code	10h	CP.1	Laptop	P
CP.3	Validating	15h	CP.2	Laptop, DBeaver, Knime	T
Phase 1	Validate Knime parent workflows and server deployment	4h	CP.2	Laptop, DBeaver, Knime	M, P
Phase 1	Implement the procedures	10h	CP.2	Laptop, DBeaver, Knime	M, P
Phase 2	Validate Web-app	1h	CP.2	Laptop	M, P
-	**Total**	**422h**	-	-	-

Table 1: Task table with the necessary duration, dependencies and resources.

The Gantt chart is included in Annex 1 at the end of the document.

For lack of time was not possible to implement the

5.4. Risk management

It is important to anticipate the risks and obstacles that may rise during development, specifically in this TFG, when using relatively new technologies, deviations from the initial planning are likely to occur. The following describes the possible obstacles and alternative plans to overcome them, and section 6.1.4 specifies the time and resources required.

1. Unforeseen difficulties - Due to the large number of tasks to be performed, the planned deliveries may not be met. In the future you can see how to manage the delivery time to be able to meet them.

2. Old programming language - Having to use MySQL which is relatively old, it does not allow you to do some functionality and may take extra time to find the expected result.

3. Database delete – As the company is reducing the budget some employee who is told to delete unused schemas could delete our database. We have to have backups from our project.

4. Lose our laptop – there is a chance that we lose the laptop, or someone steals it

6. Economic management

Once the project planning has been completed, the necessary development costs will be estimated. Different types of staff costs are identified. In addition, to overcome the obstacles that appear and assume the unscheduled costs, a contingency plan is made, a game of contingencies and mechanisms for controlling the budget are outlined.

6.1. Budget

The only have to pay me as extra budged as the server is already online.

6.1.1. Staff costs

From the planning by tasks the cost of personnel is calculated, they have take into account the 4 roles defined above: project manager, researcher, programmer and tester. Table 2 shows the hourly cost of each position, the data has been obtained from the recruitment company Hays [8].

The cost of social security is 30% of the cost of the gross salary of workers.

Role	**Cost per Hour**
Director - D	150€/h
Manager - M	25€/h
Programmer - P	15€/h

Table 2: Personnel costs from the Hays labor market guide. Own elaboration.

Table 2 details the items per task based on the costs of
staff in Table 1, and the cost of social security is estimated by multiplying the cost by 1.3.
The total cost of the project staff is € 16,435.

6.1.2. Generic costs

The temporary planning specifies that work will be done from Monday to Friday on the development of the project, and on weekends from home documenting the advanced task. Therefore, as both spaces are shared and located in Barcelona, the cost is estimated based on the rate of a coworking space[4] in Barcelona, with individual table and access every day of the week. The cost is 300 euros per month [26], includes the costs of internet, water, electricity. Considering that the project is developed over 5 months, the total cost of the space will be 1,500 euros.

The cost of the software resources used in the project is specified below, and a detailed budget is provided in Table 3.

1. DBeaver - Being an open-source tool is free.
2. Knime - Being an open-source tool is free.
3. Angular - This is an open-source tool at no cost.

Id.	Task	Time	Roles	Cost	Cost SS
PM	**Project management**	**57h**	**-**	**-**	**3113,5€**
PM.1	Scope	5h	D	750€	975€
PM.2	Planification	2h	M, P	80€	104€
PM.3	Budget	1h	D	150€	195€
PM.4	Sustainability report	6h	P	90€	117€
PM.5	Meetings	20h	D, M, P	980€	1274€
PM.6	Documentation	20h	P	300€	390€
PM.7	Presentation	3h	P	45€	58,5€
PW	**Previous Work**	**100h**	**-**	**-**	**5200€**
PW.1	Study how to develop Knime Workflows	30h	M, P	1200€	1560€
PW.2	Study of the different types of DWH	40h	M, P	1600€	2080€
PW.3	PW.3 – Study of Business Intelligence	20h	M, P	800€	1040€
PW.4	PW.4 – Study how to develop the Aktana's web-app	10h	M, P	400€	520€
DD	**DD - Development of a BI and WebApp**	**200h**	**P**	**-**	**3900€**
DD.1	Design	40h	P	600€	780€
DD.2	Implementation	120h	P	1800€	2340€
DD.3	Validation	40h	P	600€	780€

CP	**Complete Product**	**65h**	**M, P**	**-**	**3380€**
CP.1	Design	20h	M, P	800€	1040€
CP.2	Implementation	30h	M, P	1200€	1560€
CP.3	Validation	15h	M, P	600€	780€
-	**Total**	**422h**	-	**11994€**	**15593,5€**

Table 3

Since the software is all free, we will not have to spend money.

Finally, the costs of the hardware devices are calculated. Hardware costs are € 2000 for a MacBook Pro computer. Table 6 details the depreciation, the hours of use of each device are presented in the temporary planning.

For the software we have to maintain the online database which requires the maintenance of a server. We will do this through AWS by paying a monthly fee of € 5,000. If we only count the 6 months that we are going to spend doing the project, we get € 30,000 for the server.

6.1.3. Contingency

As with any project, it is important to add an extra cost to cover obstacles and contingencies. In this case, as it is a research project with innovative technologies, the probability of encountering problems during development is considerable, therefore, it has been decided to set a 15% additional cost. Table 4 details the total contingency of the project.

Type	**Cost**	**Contingency**
Space	1.500€	225€
Software	30000€	4500€
Staff	16435€	2465€
Total	**17935€**	**7190€**

Table 4

[4]Coworking space: Shared spaces where the independent professionals develop their projects

6.1.4. Unforeseen

Finally, the cost of obstacles that may arise during is calculated
project development. The contingencies are presented in the planning schedule, below:

1. Increased development time - If you need more development time, 25 hours of development will be added to the planning and 10 hours of testing. The total cost would be 25 hours of programmer and 10 hours of tester, therefore, 560 euros. The risk is high due to the use of new technologies, so an estimated 20% probability.

6.1.5. Total cost

Once all the costs of the project have been presented, Table 5 presents the final budget for the work. The total cost of the project is 53,319 euros.

Type	**Cost**
Space	1.500€
Hardware	2.000€
Software	30.000€
Staff	15593,5€
Contingency	2.690€
Unforeseen	544€
Total	**52327.5€**

Table 5

6.2. Management control

Once the initial budget is defined, the control mechanisms are defined necessary to avoid deviations, as well as numerical indicators that help to control. At weekly meetings, each time a task is completed, the budget will be updated with the actual hours and compared with the estimated hours.
To control contingencies, at the end of a task, the extra expenses that have been incurred will also be noted and will be compared with the contingency forecast and contingency. In this way, you can quickly detect any deviations and predict whether you need to cut back on a task or increase your budget.

The following are the numeric descriptors for the control:

1. Deviation of personal cost per task:
(Estimated cost - actual cost) ∗ actual hours

2. Deviation from performing tasks:
(Estimated hours - actual hours) ∗ actual cost

3. Total deviation in the performance of tasks:
total estimated cost - total actual cost

4. Total deviation of resources (software, space or staff):
total estimated cost - total actual cost

5. Total Deviation Cost of Contingencies: Estimated Cost of Contingencies - Actual Cost of Contingencies

6. Total deviation of hours: estimated hours - actual hours

7. Sustainability

In any project it is important to perform a sustainability analysis considering three dimensions: economic, environmental and social. Then the author of the EOD conducts a self-assessment on the domain of competition sustainability, and then, based on a few questions, the three dimensions in the framework of the project.

7.1. Self-assessment

Throughout the Degree in Computer Engineering, we have been told in what is sustainability, as well as the importance of thinking about the environment every time we do a project. However, until we do a job like the EOD, we don't realize how important it is.
From a personal point of view, I have always kept in mind the need for an economic analysis to ensure the viability of a project, as work often seeks to improve the economic aspect of an existing solution.

7.2. Economic Dimension

Regarding the cost of carrying out the project, considering that the use of this will be for managers to make decisions regarding the management of employees, I think that the cost of the project is adequate as that can significantly increase the management of company resources.

7.3. Environmental Dimension

Fortunately, the realization of this project does not have a big environmental impact, only the manufacture of the computers to be used has a negative impact on the environment. The Amazon Web Services are almost fully powered by renewable resources reducing the server's environmental impact.

The only way to improve your environmental impact is to find a computer that meets the best environmental requirements. You could also look for which of the best possible programs and programming languages to use the project are the most efficient and environmentally friendly.

7.4. Social Dimension

There is no aspect that can be said to be socially beneficial as it is a project aimed at a particular company in a very closed environment where only the company and the people who work in this will benefit.

8. Justification

I have chosen this project because it was a company need from long time ago and now that I had the opportunity to develop my EOD I thought it was the perfect time to develop it as I think it can be very useful in the development of my professional career since it is a very good learning exercise as I must make the best decision by myself always considering the requirements of the company. In addition, I will have to learn to manage a whole set of tools and put all the concepts in common to be able to close a consistent and solid project.

Also, I wanted to make this work as it's an opportunity to make relations with the high executives in the company and this can usefully in a future to know how to interact with them as an extra knowledge.

This project will help the company since by using this platform the company will be able to have a higher performance and the possibility of increasing economic profits. As well, will free up lot of time to the operations director which is a position where the time is gold in order to be able to take better decisions as he is going to have extra time to deliberate making with this a better company performance.

9. Tools

For the development of the application a set of software and hardware tools were necessary to be learnt and be capable of solving problems with these. Also, I needed to learn how the employees where reporting the hours and understand from where the best side was to get the information in order to have the maximum consistency and at the same time getting the largest amount of data as possible.

9.1. Software

The initial system was created in a database by using MySQL language, which allows us to develop relational databases. The server MySQL user version is 5.7.33. It's a bit legacy but we must develop the project on that as the company is developed under this language version and changing a whole company language would be hard.

9.1.1. MySQL

MySQL is a relational database management system (RDBMS) developed by Oracle that is based on structured query language (SQL).

MySQL is integral to many of the most popular software stacks for building and maintaining everything from customer-facing web applications to powerful, data-driven B2B services. Its open-source nature, stability, and rich feature set, paired with ongoing development and support from Oracle, have meant that internet-critical organizations such as Facebook, Flickr, Twitter, Wikipedia, and YouTube all employ MySQL backends.

Image 2 - MySQL

Though often associated with internet applications or web services, MySQL was designed to be extensively compatible with other technologies and architectures. The RDBMS runs on all major computing platforms, including Unix-based operating systems, such as the myriad Linux distributions or Mac OS, and Windows.

The primary factor differentiating relational databases from other digital storage lies in how data is organized at a high level. Databases like MySQL contain records in multiple, separate, and highly codified tables, as opposed to a single all-encompassing repository, or collections of semi- or unstructured documents.

This allows RDBMSs to better optimize actions like data retrieval, updating information, or more complex actions like aggregations. A logical model is defined over all of the contents of the database, describing for example the values allowed in individual columns, characteristics of tables and views, or how indices from two tables are related.

Relational models have remained popular for several reasons. They empower users with intuitive, declarative programming languages — essentially telling the database what result is wanted in language akin to, or at least comprehensible as, written english, instead of

meticulously coding up each step of the procedure leading to that result. This moves a lot of the work into the RDBMS and SQL engines, better enforcing logical rules and saving valuable resources and manpower.

9.1.2. Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.
For our case python was used to implement certain Knime nodes as is had to be done by an script as the internal workflow nodes libraries were failing.

Image 3 - Python

9.1.3. AWS

For the server maintenance will be used AWS. Amazon Web Services offers a broad set of global cloud-based products including compute, storage, databases, analytics, networking, mobile, developer tools, management tools, IoT, security, and enterprise applications: on-demand, available in seconds, with pay-as-you-go pricing. From data warehousing to

deployment tools, directories to content delivery, over 200 AWS services are available. New services can be provisioned quickly, without the upfront fixed expense. This allows enterprises, start-ups, small and medium-sized businesses, and customers in the public sector to access the building blocks they need to respond quickly to changing business requirements. This whitepaper provides you with an overview of the benefits of the AWS Cloud and introduces you to the services that make up the platform.

Image 4 – Amazon Web Service

9.1.4. KNIME

For the development of the ETL workflows we will use Knime3 [4].
KNIME (or Konstanz Information Miner) is a data mining platform that enables model development in a visual environment. It is built under the Eclipse platform.
KNIME Analytics Platform is the open source software for creating data science. Intuitive, open, and continuously integrating new developments, KNIME makes understanding data and designing data science workflows and reusable components accessible to everyone.
Since 2006, KNIME has been used in pharmaceutical research, it also used in other areas such as CRM customer data analysis, business intelligence, text mining and financial data analysis. Recently attempts were made to use KNIME as robotic process automation (RPA) tool.
We are going to use Knemi just for the integration of the data coming from the APIs since we are making all the transformations inside the database.
Was decided to use this tool for the integration part as is the tool which is using the company's data science team.

Image 5 - Knime

9.1.5. TimeCamp API

TimeCamp offers employee time tracking with project management features. Its unique project tree structure allows you to add unlimited projects and tasks, track and manage time, and then generate the reports for time analysis.

Work tracking provides project managers with a comprehensive analysis of how team members spend their work hours across their assignments. It sounds quite tedious, but luckily, the market is full of various SaaS work time tracking apps that streamline logging work hours.

For this project will be needed to be used the TimeCamp API as its from where we are going to import most of the data.

Image 6 – TimeCamp

9.1.6. Google Sheets API

Google Sheets is a spreadsheet program included as part of the free, web-based Google Docs Editors suite offered by Google. The service also includes Google Docs, Google Slides, Google Drawings, Google Forms, Google Sites and Google Keep. Google Sheets is available as a web application, mobile app for: Android, iOS, Microsoft Windows, BlackBerry OS and as a desktop application on Google's Chrome OS. The app is compatible with Microsoft Excel file formats.[2] The app allows users to create and edit files online while collaborating with other users in real-time. Edits are tracked by user with a revision history presenting changes. An editor's position is highlighted with an editor-specific colour and cursor and a permissions system regulates what users can do. Updates have introduced features using machine learning, including "Explore", offering answers based on natural language questions in a spreadsheet.

Image 7 – Google sheet

9.1.7. DBeaver

DBeaver is a SQL client software application and a database administration tool. For relational databases it uses the JDBC application programming interface (API) to interact with databases via a JDBC driver. For other databases (NoSQL) it uses proprietary database drivers. It provides an editor that supports code completion and syntax highlighting. It provides a plug-in architecture (based on the Eclipse plugins architecture) that allows users to modify much of the application's behaviour to provide database-specific functionality or features that are database-independent. This is a desktop application written in Java and based on Eclipse platform.

Image 8 – DBeaver

9.1.8. Excel

Microsoft Excel is a spreadsheet developed by Microsoft for Windows, macOS, Android and iOS. It features calculation or computation capabilities, graphing tools, pivot tables, and a macro programming language called Visual Basic for Applications (VBA). Excel forms part of the Microsoft Office suite of software. Excel was used to see the imported data in a visual way.

Image 9 - Excel

9.1.9. Jira Kanban

Jira Software provides planning tools and roadmaps so teams can manage stakeholders, budgets, and role requirements from day one.
Kanban is a common framework for agile software development and DevOps that provides transparency of work and team capacity.
A kanban board is one of the tools that can be used to implement kanban to manage work at a personal or organizational level.
Simple boards have columns for "waiting", "in progress" and "completed" or "to-do", "doing", and "done". Complex kanban boards can be created that subdivide "in progress" work into multiple columns to visualise the flow of work across a whole value stream map.

Image 10 - Jira

9.1.10 Tableau

Tableau Software is a software company headquartered in Seattle, United States, which develops interactive data visualization products that focus on business intelligence. It began marketing research, which was done in the department of computer science at Stanford University from 1999 to 2002. It was founded in Mountain View, California in January 2003 by Chris Stolte, who specializes in visualization techniques for exploring and analyzing relational databases and data cubes. Christian Chabot and Pat Hanrahan combined a structured database search language with a descriptive language to represent graphics, as well as inventing a visual database language called VizQL. VizQL formed the core of the Polaris system, an interface to explore large, multidimensional databases. The product searches within relational databases, OLAP cubes, databases in the cloud and spreadsheets and then a number of chart types are generated.

Tableau contains map functionality, and any latitude and longitude coordinates can be plotted. It has been criticized for being too focused on the United States. They offer custom geocoding as well as five ways to access their products: Desktop (professional and personal editions), server, online, reader, and public. The latter two are free for the user.

Image 11 - Tableau

9.2. Hardware

9.2.1 Laptop

We need a machine to work in this project. I am using a MacBookPro15,2, Quad-Core Intel Core i7, 2,8 GHz, 16 GB.

Image 12 - Laptop

10. Business Inteligencie Development

10.1. Implementation

In this section, each of the tasks that have been planned to will be developed.
In the following section I will comment the project's implementation while commenting on some of the obstacles encountered and the solution chosen for them.
This part has the largest weight in the project as it is the phase where it is going to be made al the project building.

10.1.1. Staging Tables

Before developing the Workflows was needed to create all the staging tables. We want to store the historic of all the record as it is needed for the development of the Data Warehouse. For doing that the staging tables had to be created. These are the tables populated by the data coming from the ETL. These tables are key on the development of the DWH. The tables have a valid_from and valid_to fields which will be used to know the period of time where a row is valid. For having a historical record of the row was necessary to created two layers of staging tables.

- The first layer will be the non-historical layer. This tables will contain the most updated information as this tables are populated directly from the ETL. In these tables are implemented trigger which are going to be called every time the ETL inserts or updates a row to each of the non-historical tables. These triggers when are called, insert the lasts updated or inserted rows into historical staging tables by initializing the "valid_from" field as "current_date" and the "valid_until" as "infinity". The "valid_from" field from the already existing id is changed to "current_date -1" because the entry is no longer valid.

- The historical ones are never updated, they are always being inserted. These tables are populated by the nonhistorical staging table triggers. As I have explained in the previous point, when a row is inserted in the nonhistorical staging table, a trigger is executed. Before doing it the field valid_until from the historical staging tables has to be modified for having the current date as today is the last day this row is valid. For now and ahead, the valid row is the one it was imported in the last update. By doing this, it is achieved to have the historical record of all the data coming from the API stored every time a changes is made.

10.1.2. Data integration

The first, and one of the most important steps, was importing all the needed data from TimeCamp source into our database staging tables.
The fountains where we were importing data are TimeCamp and Google Sheet.

- From TimeCamp, was from where we imported all information related to the USER, Hours reported by each user for each task, the GROUP each user belongs, and the reported task id.
  The triggers can be checked in the Annex form 2 to 7.

- Also was imported data from a google sheet where the director was updating weekly. There, is all the employee information related. When a change is done to a user, for example change role, it is being reported in this google sheet.

For importing all the necessary data we used Knime. This tool consists in developing several workflows, each one retrieving data from the corresponding API, transforming the coming data, which could be XML or JSON format and inserting into our database staging tables.

This are the different implemented Knime workflows, we are going to delve in each workflow. The workflows are composed of nodes. A node is a Knime structure that allows you to make data operations. The nodes must be configured previously, each node has his own configuration. Once finished the configuration these are linked in the proper way.

Now we are going to see each workflow in more detail:

This first workflow is in charge of loading, updating and deleting the "STG_TIMECAMP_ENTRIES" table from our database.

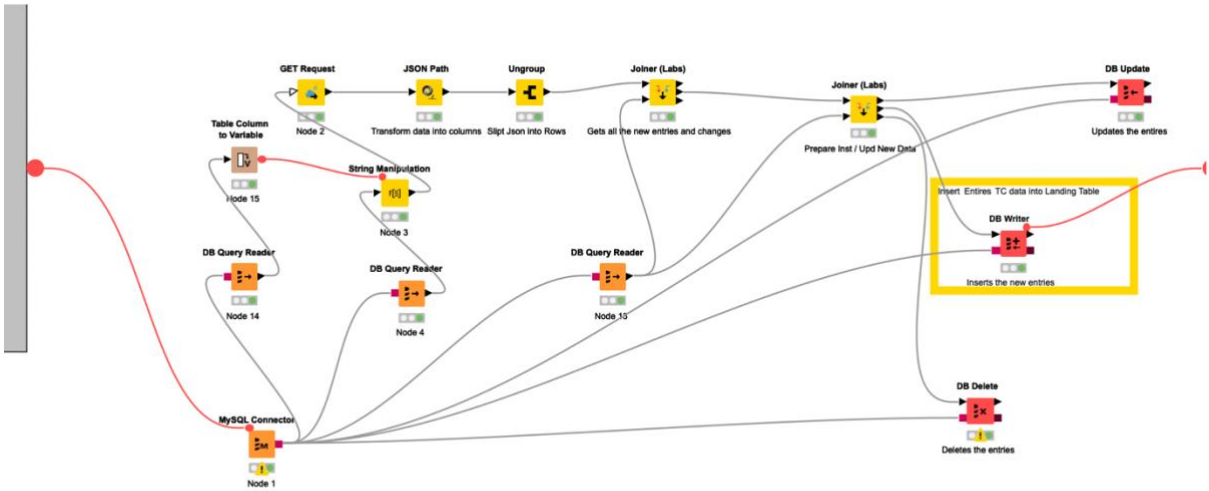


Image 13 – Entries Knime Workflow

The first node will able us to connect to the database, then the three query readers will retrieve the desired data from our DB, the right one will get the staging table by executing "SELECT DATE_SUB(NOW(), INTERVAL 90 DAY)" query. The other two readers will get the corresponding metadata to build the URL. In the "String Manipulator" node is built the URL as we see in the Figure X. In the case of this workflow, we need to load the data by using and interval of time as the full TimeCamp entries data is too large to be loaded every week and it also was making Knime crash. For the URL there it is also added the API token.

Once builded the URL is inserted into the "GET Request" node, so this is able to return the data from Time Camp API. The following two nodes convert the Entries Time Camp data into tables as it is coming in JSON format. The first "Joiner" gets all the non-stored data in our database by joining the data base and API tables by the hash. The hash is the concatenation of all the table fields so, we can detect when a change is done. In other words the new data that has to be updated.

In the last joiner, the data is filtered by ID and splitting the output by Joined rows, left unmatched rows and right unmatched rows. If the ID matches, means the entry already exists in the DB but now updating it with the latest data, so we are going to call the "DB Update" node to update the corresponding row. The rest of the rows returned are the unmatched ones. These rows have to be inserted or deleted. The rows coming from the left output are the ones existing in the API but not in our DB so these have to be inserted. For last, the right unmatched rows have to be deleted as we have it the DB but are no longer in the API.

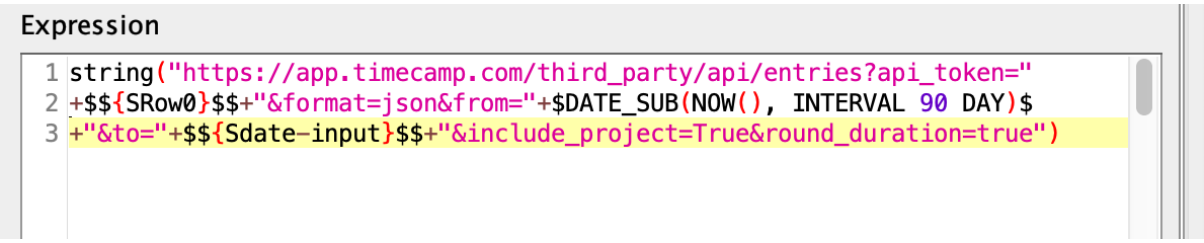


Image 14 – Knime Entries URL

The rest of the TimeCamp workflows follow the same structure with little changes in their structure.

Group Workflow:

This workflow gets the TimeCamp Group data from the API, transform into tables the data and inserts it into the STG_TIMECAMP_GROUP table from the database.

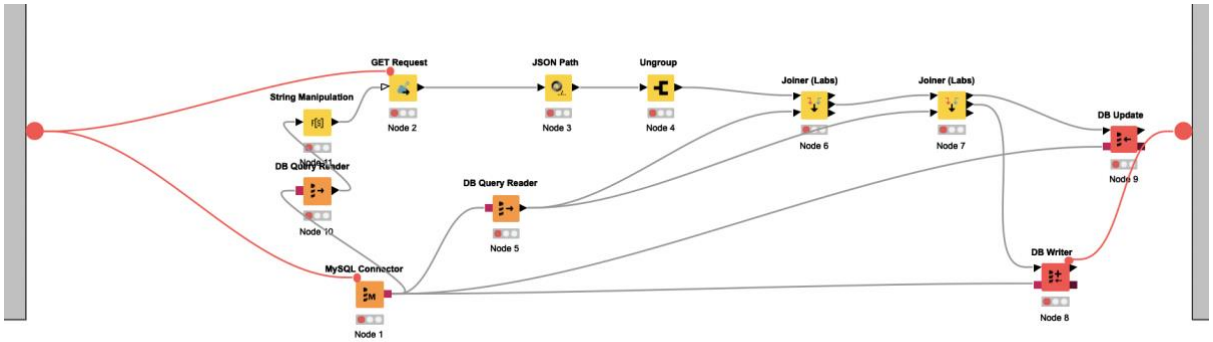


Image 15 – Knime Group Workflow

The first node connects to our database, then the data is retrieved by the "Query Readers" nodes. To build the rest of the TimeCamp URLs it is not need the data filter as the API URL for the Group data doesn't request it, so we can build it just with the API token like in the Figure !, and we will be retrieving all the data every time from the API.

Image 16 – Knime Grop URL

Once the URL is generated, as before, it is inserted into the "GET Request" node, the next nodes transform the data into tables, the new data is returned by the first join. In this case there is no need to delete the rows, so the data is updated or inserted after passing through the last joiner.

User Workflow:

This workflow gets the TimeCamp User data from the API, transform into tables the data and inserts it into the STG_TIMECAMP_USER table from the database.

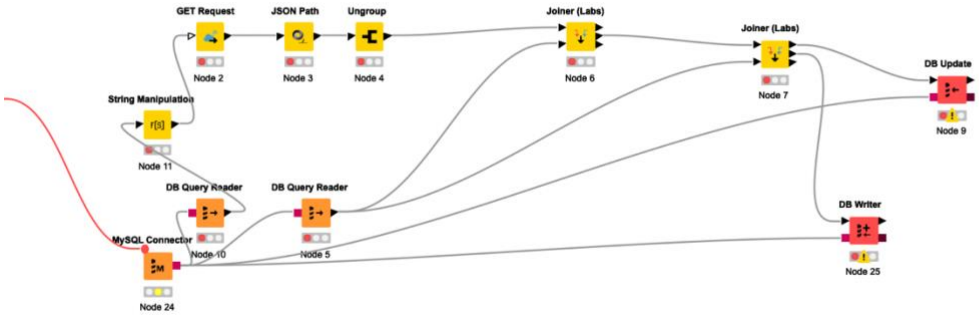


Image 17 – Knime User Workflow

The nodes have the same configurations as the TimeCamp Group ones. The only different parts are the Query Reader node that now it has to read the data from STG_TIMECAMP_GROUP table instead of the Entres one, from the DB, and change the update and writer nodes to now write into the STG_TIMECAMP_GROUP table.

Task Workflow:

Same as before, the nodes have the same configs, and the changes are that now the data is retrieved and inserted into the Task staging table.
This workflow gets the TimeCamp Task data from the API, transform into tables the data and inserts it into the STG_TIMECAMP_USER table from the database.

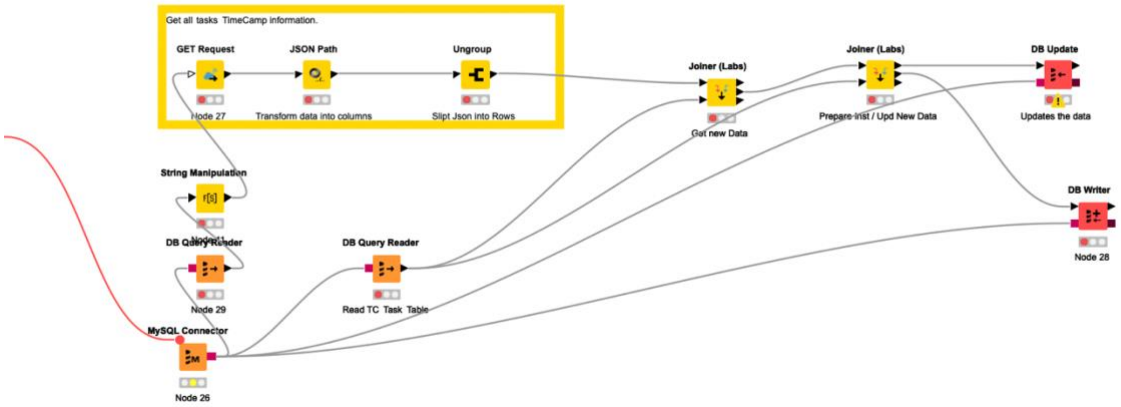


Image 18 – Knime Task Workflow

Employee Workflow:

This last workflow is a little different as the ones before. Now the data is coming from a google sheet instead of the TimeCamp API. Google library had to be imported
to get the data from there.

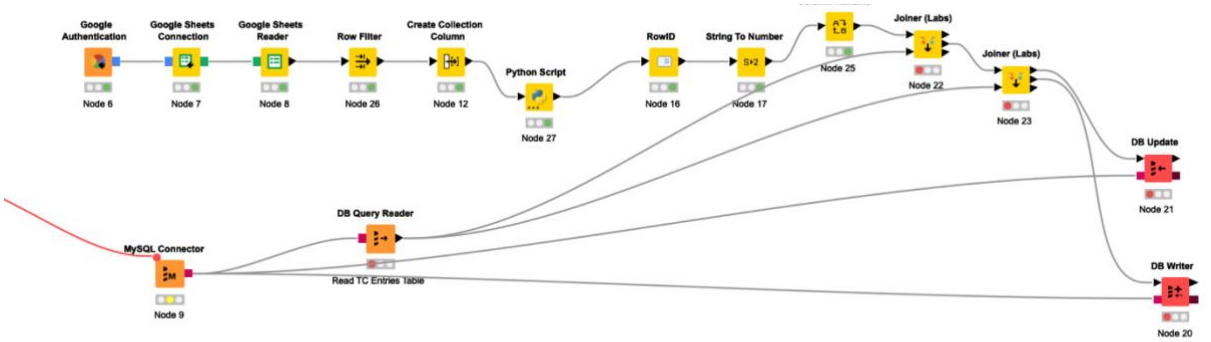


Image 19 – Knime Employee Workflow

First of all, as it has been done in the rest of the before nodes the connector node is connected into the DB and the query reader reads the STG_EMPLOYEE table.

To start getting the data from the Google Sheet there is needed a google authentication permission that has to be granted by the file creator. Once given we connect with our credentials and save it to the node. Afterwards "Google Sheet Connections" node connects to the sheet and the "Google Sheets Reader" returns the file data. The empty rows are cleaned by the "Row Filter". Subsequently, a new column is created which will be the one containing the hashkey which will be generated by the Python Script which you can see in the Figure z.

Later, the data is transformed into tables and filtered by hashkey then by id and update the DB, as it has been done in the rest of the nodes before.

```
# Copy input to output
import pandas as pd
input_table_1['hashkey'] = pd.DataFrame(input_table_1['hashkey'])
input_table_1['hashkey'] = [str(x) for x in input_table_1['hashkey']]
input_table_1['hashkey'] = [x.replace("'","") for x in input_table_1['hashkey']]
input_table_1['hashkey'] = [x.replace(" ","") for x in input_table_1['hashkey']]
output_table_1 = input_table_1
print(output_table_1['hashkey'][0])
```

Image 20 – Knime Python node script

10.1.3. Knime Parent Workflow Development and Deployment

To end with the data integration part, there is the main workflow which is composed of metandoes[5]. Each of this metanodes contains one of the corresponding workflows explained before. This are executed sequentially, for not having consistency errors, by using the Flow Variable[6] initialized in the "Date&Time Configuration" node. Once all the metanodes have ended the execution the Python script calls the Parent Procedure in our database, and it will update all the DIMs and FACTs.

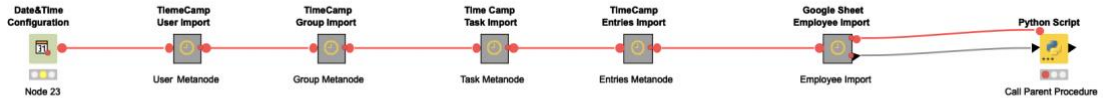


Image 21 – Knime Parent Workflow

To end, the main workflow has been deployed in Aktana's knime server located in San Francisco, California.
As we are having as an objective to the automatization of the BI, which means we don't want to be pending of running every week the job, the execution is scheduled periodically for every Monday at 9:30 AM (GMT) by using the internal Knime clock. This time was set like this as our boss wanted to have updated all the last week information.

[5]Metanode: A metanode is a Knime structure that contains more nodes inside it.
[6]Flow variable: Knime Global variable that allows to control the node flow

10.2. Data transformation

In this stage of the project, we are going to develop the Data Warehouse. To make it, it's necessary many types of tables and procedures. In the following image x there is all the DWH needed table.

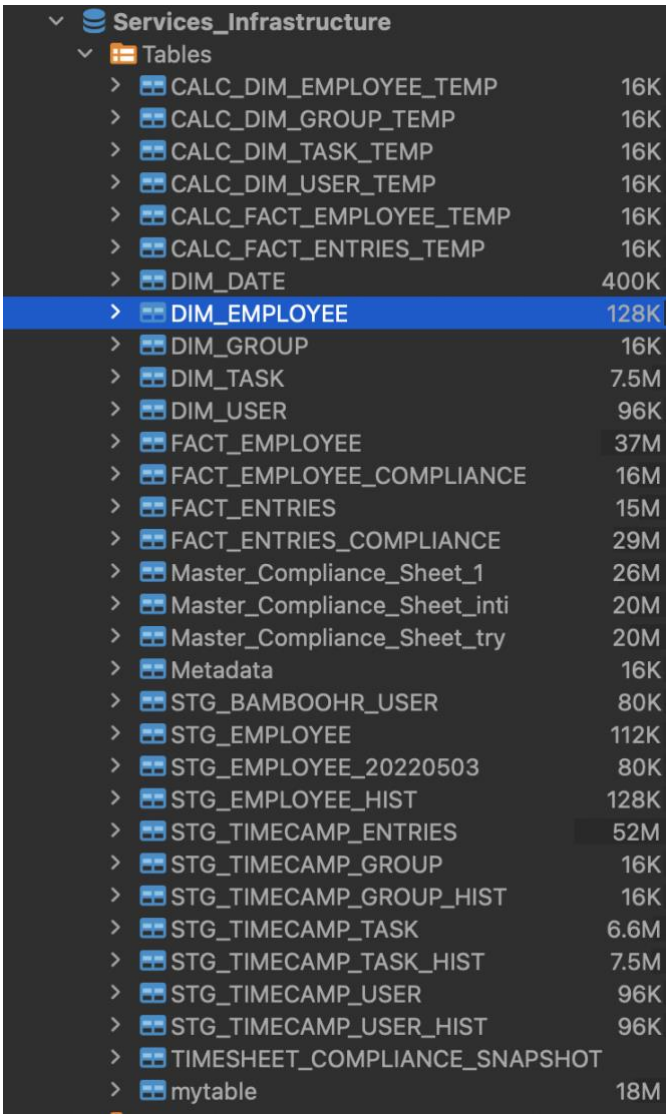


Image 22 – Full database schema

10.2.1. DWH explanation

A data warehouse is a type of data management system that is designed to enable and support business intelligence (BI) activities, especially analytics. Data warehouses are solely intended to perform queries and analysis and often contain large amounts of historical data. The data within a data warehouse is usually derived from a wide range of sources such as application log files and transaction applications.

A data warehouse centralizes and consolidates large amounts of data from multiple sources. Its analytical capabilities allow organizations to derive valuable business insights from their data to improve decision-making. Over time, it builds a historical record that can be invaluable to data scientists and business analysts. Because of these capabilities, a data warehouse can be considered an organization's "single source of truth."

To introduce in DWH is needed to know that the data will be transformed to make queries as efficient as possible as we are managing large amount of data. To achieve that it is necessary to split the data into dimensions (DIM). A dimension is a sort of table that will be populated with all the data coming from the historical stage table, then a surrogate key[7] will be generated for each new row inserted making unique each row of the DIM.

On the other hand, we have the Fact tables (FACT). The fact tables are a sort of tables where we are going to have stored all the DIMs surrogate keys with the desired metrics.

Making this, we have all the calculated information (metric) in the fact table and all the attributes in the DIMs so when we want to make a query we don't need to go all around the tables.

[7]Surrogate Key: Is a unique identifier for either an entity in the modelled world or an object in the database

10.2.2. DWH modelling.

To begin, the first data transformation is done in the historical staging tables once this are loaded.

To develop the DWH was needed to think about which sort of model was the most optimal to implement and how was wanted to be done. Our DWH uses a OLAP[8] model. The DWH was needed to have a dimension for each source of data we were having in our staging tables.

Three Fact tables were requested by the director:

- The FACT Entries
- The FACT Entries Compliance
- The FACT Employee.

Looking into each fact purposes: the first FACT purpose is to have all the actual entries data linked to the corresponding DIMs which will be linked to all.
The second one, the FACT Entries Compliance has the purpose to know the Employees weekly Compliance. It means we want to have a record week by week about how the entries table was updated. By doing this we can know which employees report the hours when it's supposed to be done and if they change the entries reports the next week.
For last one there is the Fact employee, this has records of all the active employees from the company start. This let us know the company's employee behaviour and the company's growth by the time.

To make work the facts, as we said, DIMs are needed. Five DIMs were needed to be created.

- DIM User (contains all the attributes related with the TimeCamp User created)
- DIM Employee (contains all the attributes related to the Google Sheet employee)
- DIM Group (contains all the attributes related with the TimeCamp Group)
- DIM Task (contains all the attributes related with the TimeCamp Task)
- DIM Date (contains all the dates, week name, etc… from 2016)

In the figure x is visible an UML schema of the DWH final model.

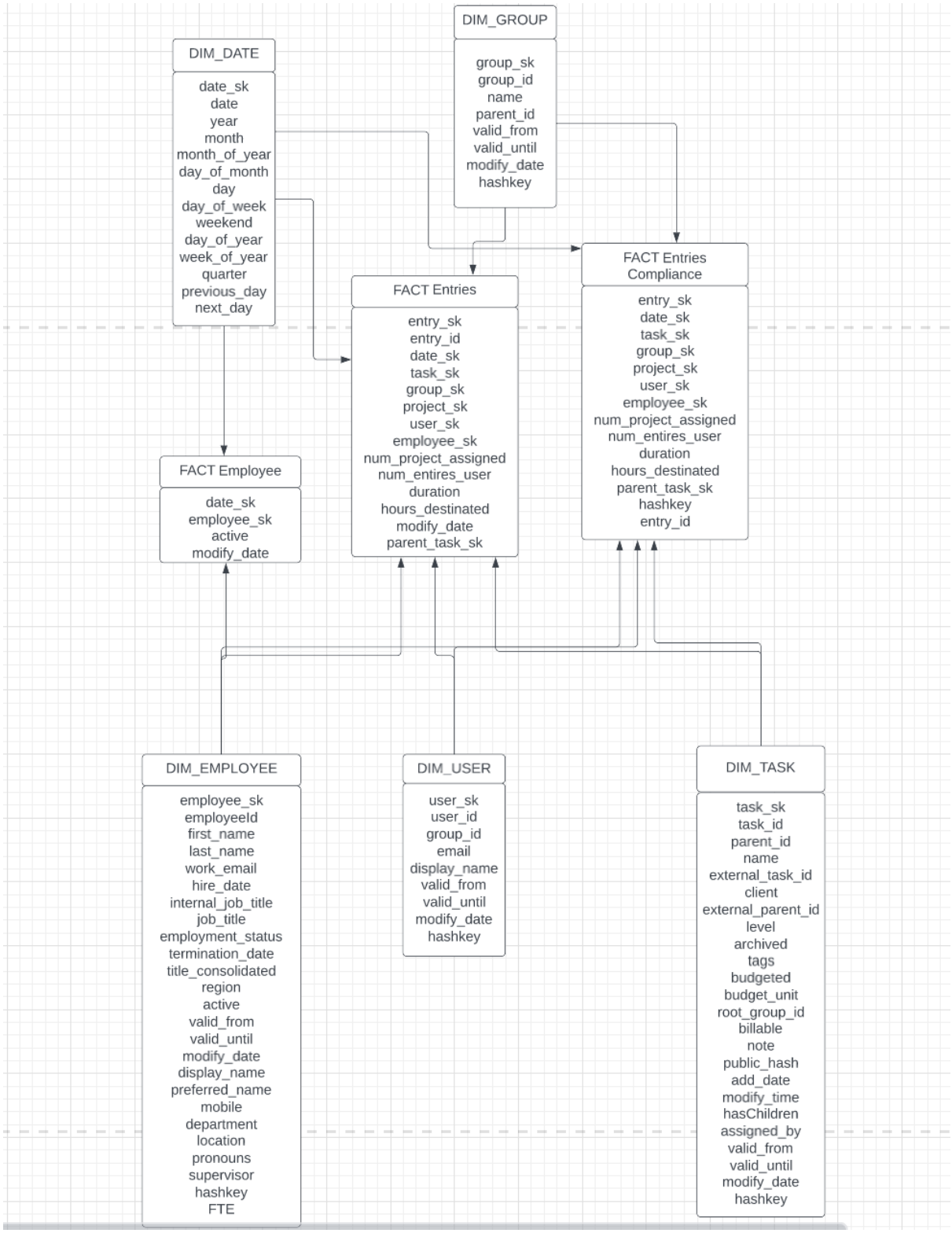


Image 23 – DWH UML diagram

[8]Online Analytical Processing Server (OLAP): is based on the multidimensional data model. It allows managers, and analysts to get an insight of the information through fast, consistent, and interactive access to information.

10.2.3. Data Updating

The data from the FACTs and the DIMs will be updated by using procedures. Here, in the figure x there are all the procedures created in our Database. For the proper usage of the DWH only the INSERT_FACT_EMPLOYEE_COMPLIANCE, UPSERT_DIM_EMPLOYEE, UPSERT_DIM_GROUP, UPSERT_DIM_TASK, UPSERT_DIM_USER, UPSERT_FACT_ENTRIES, UPSERT_FACT_EMPLOYEE, PARENT_PROCESS_DWH.

The other insert procedures are not needed for the Database update, they are created for initializing the DIMs easier in case these are deleted.
The upsert procedures are in charge of updating and inserting the DIMs and FACTs.

Finally, all the procedures are gathered in the PARENT_PROCESS_DWH workflow. This workflow calls the rest of needed workflow.

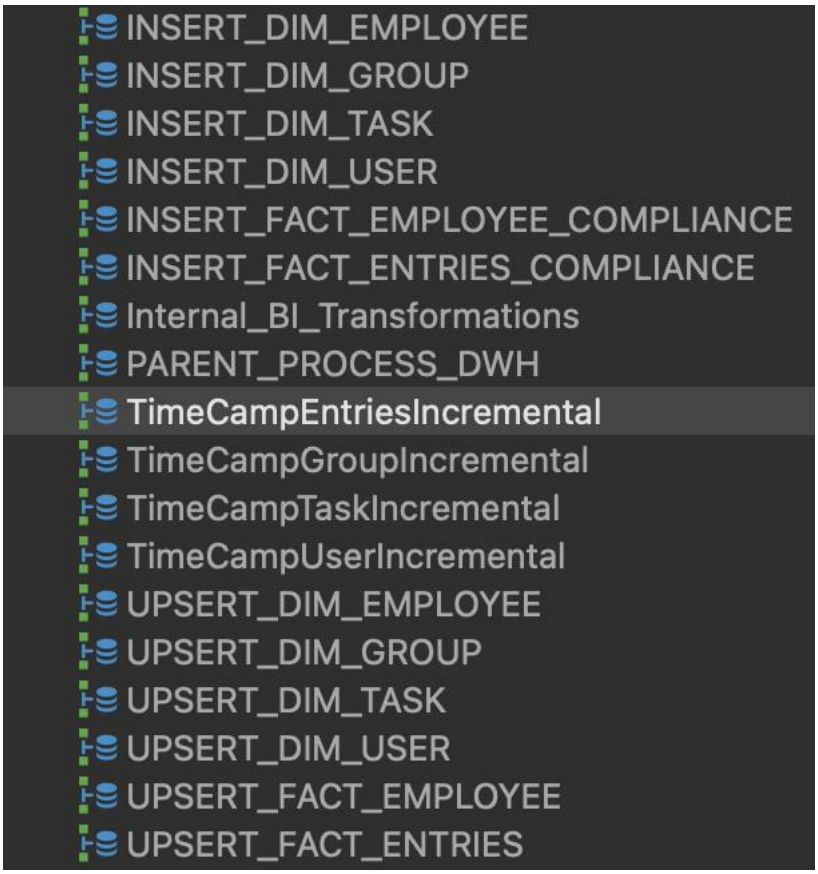


Image 24 – DWH procedures

Going deeper in each procedure to explain in a high level what these are suposed to be doing in the DWH.

- INSERT_DIM_EMPLOYEE: The purpose of this procedure is to load form zero the DIM_EMPLOYEE if its needed because for some reason the DIM was deleted or truncated.

This procedure gets the data from the STG_EMPLOYEE_HIST historical staging table and identically inserts all the data in the corresponding fields.

- INSERT_DIM_USER: The purpose of this procedure is to load form zero the DIM_USER if its needed because for some reason the DIM was deleted or truncated. This procedure gets the data from the STG_TIMECAMP_USER_HIST historical staging tables and identically inserts all the data in the corresponding fields.

- INSERT_DIM_GROUP: The purpose of this procedure is to load form zero the DIM_GROUP if its needed because for some reason the DIM was deleted or truncated. This procedure gets the data from the STG_TIMECAMP_GROUP_HIST historical staging tables and identically inserts all the data in the corresponding fields.

- INSERT_DIM_TASK: The purpose of this procedure is to load form zero the DIM_TASK if its needed because for some reason the DIM was deleted or truncated. This procedure gets the data from the STG_TIMECAMP_TASK_HIST historical staging tables and identically inserts all the data in the corresponding fields.

- UPSERT_DIM_EMPLOYEE: The purpose of this procedure is to update and insert (upsert) the DIM_EMPLOYEE. We wanted to have a good performance of the Data Warehouse and to do that we wanted to use CTE[8] but was not possible as we are using a legacy MySQL verion and it doesn't support these structures. In order to make it the most efficient as possible, we have created a sort of temporal tables which will be truncated after the procedure execution. This certain procedure uses the CALC_DIM_EMPLOYEE_TEMP temporal table.
  The data from the STG_EMPLOYEE_HIST is loaded into the temporal table. There, it is selected the rows which are going to be updated and which rows are going to be inserted. Once finished the insertion, the DIM EMPLOYEE has been updated successfully and the temporal table will be truncated.

- UPSERT_DIM_USER: The purpose of this procedure is to update and insert (upsert) the DIM_EMPLOYEE. As explained in the UPSERT_DIM_EMPLOYEE procedure, we will be using temporal tables. This certain procedure uses the CALC_DIM_USER_TEMP temporal table.
  The data from the STG_USER_HIST is loaded into the temporal table. There, it is selected the rows which are going to be updated and which rows are going to be inserted. Once finished the insertion, the DIM USER has been updated successfully and the temporal table will be truncated.

[8]CTE: A common table expression (CTE) is a named temporary result set that exists within the scope of a single statement and that can be referred to later within that statement, possibly multiple times.

- UPSERT_DIM_GROUP: The purpose of this procedure is to update and insert (upsert) the DIM_GROUP. As explained in the UPSERT_DIM_EMPLOYEE procedure, we will be using temporal tables. This certain procedure uses the CALC_DIM_GROUP_TEMP temporal table.
  The data from the STG_GROUP_HIST is loaded into the temporal table. There, it is selected the rows which are going to be updated and which rows are going to be inserted. Once finished the insertion, the DIM GROUP has been updated successfully and the temporal table will be truncated.

- UPSERT_DIM_TASK: The purpose of this procedure is to update and insert (upsert) the DIM_EMPLOYEE. As explained in the UPSERT_DIM_TASK procedure, we will be using temporal tables. This certain procedure uses the CALC_DIM_TASK_TEMP temporal table.
  The data from the STG_TASK_HIST is loaded into the temporal table. There, it is selected the rows which are going to be updated and which rows are going to be inserted. Once finished the insertion, the DIM TASK has been updated successfully and the temporal table will be truncated.

- UPSERT_FACT_ENTRIES: The purpose of this procedure is to upsert the FACT_ENTRIES table. To do that in a efficient way I am still going to be using the temporal tables. In this case, the CALC_FACT_ENTRIES_TEMP temporal table will be used.
  To start with the procedure, it is going to be loaded the temporal table with the newest data from the STG_TIMECAMP_ENTRIES being joined by key with the DIMs. With that we get all the latest surrogate keys in the temporal fact table. Once the temporal table is populated it is needed to insert the latest information into the FACT_ENTRIES table and update the corresponding rows with the newest data. Once finished the temporal table will be closed.

- UPSERT_FACT_EMPLOYEES: The purpose of this procedure is to upsert the FACT_EMPLOYEES table. To do that in a efficient way I am still going to be using the temporal tables. In this case, the CALC_FACT_EMPLOYEES_TEMP temporal table will be used.
  To start with the procedure, it is going to be loaded the temporal table with the newest data from the STG_ EMPLOYEES being joined by key with the DIMs. With that we get all the latest surrogate keys in the temporal fact table. Once the temporal table is populated it is needed to insert the latest information into the FACT_EMPLOYEES table and update the corresponding rows with the newest data. Once finished the temporal table will be truncated.

- INSERT_FACT_ENTRIES_COMPLIANCE: The purpose of this procedure is to upsert the FACT_ENTRIES_COMPLIANCE table. This table has a different property from the before ones. This table is needed to be populated only from weekly inserts. This means that we will only need to insert in this table, never update. To do that in a efficient way I am still going to be using the temporal tables. In this case, the CALC_FACT_ENTRIES_TEMP temporal table will be used.
  This procedure, as we told before, needs to only insert the previous week. To make it possible I added a date filter, to only get the previous week from Monday to Sunday and it is going to be loaded into the temporal table with the newest data from the STG_TIMECAMP_ENTRIES being joined by key with the DIMs. With that we get all the latest surrogate keys in the temporal fact table. Once the temporal table is populated it is needed to insert the latest information into the FACT_ENTRIES_COMPLIANCE table and update the corresponding rows with the newest data. Once finished the temporal table will be closed.

10.2.4. Load FACT_ENTRIES_COMPLIANCE from an existing excel

This part was not planned to do but was a request from the director of the project. It consisted about loading an Excel where they were recording the entries compliance weekly since the year 2020. This part took me a lot of time as the date was coming with Chinese characters in the description and name fields causing issues. In the end I opted to delete these fields and importing the file without.

For the file imports I had to find a way to detecting the fields and having the opportunity to delete it. The file was too large, it was containing more than 100,000 rows and I was not able to see what was DBeaver doing while importing the file. I tried so many times and I couldn't import the file successfully. Finally I found a internet platform[27] where I could detect and remove the fields for later translating it into MySQL. I executed all the generated inserts and the files was successfully imported.

To finish with the FACT_ENTRIES_COMPLIANCE loading was necessary to link the imported file with the DIMs information. I executed the following query, the one that loads the FACT_ENTRIES_COMPLIANCE but there were still missing so many rows.

```
insert into FACT_ENTRIES_COMPLIANCE
        SELECT
                0,
                d.date_sk  ,
                tsk.task_sk,
                grp.group_sk,
                entr.project_id,
                usr.user_sk ,
```

```
                emp.employee_sk ,
                0,
                0,
                entr.duration,
                entr.duration,
                tsk.parent_id,
                null,
                entr.id
                FROM mytable entr
                JOIN DIM_DATE  d ON convert(d.date_sk,date) = entr.`date`

                JOIN DIM_USER usr ON entr.user_id = usr.user_id  AND entr.`date`
>= usr.valid_from  and entr.`date` <= usr.valid_until
                JOIN DIM_GROUP grp ON grp.group_id = usr.group_id AND
entr.`date` >= grp.valid_from AND entr.`date` <= grp.valid_until
                JOIN DIM_TASK tsk ON entr.task_id = tsk.task_id  AND entr.`date` >=
tsk.valid_from AND entr.`date` <= tsk.valid_until
                -- JOIN DIM_TASK TSK_P on tsk.parent_id = TSK_P.task_id  AND
entr.`date` BETWEEN TSK_P.valid_from AND TSK_P.valid_until
                -- JOIN Services_Infrastructure.DIM_EMPLOYEE  emp ON
UPPER(emp.work_email) = UPPER(concat(entr.user_name ,"@aktana.com")) AND
entr.`date` >= emp.valid_from and  entr.`date` <= emp.valid_until
                JOIN DIM_EMPLOYEE emp on upper(emp.work_email) =
UPPER(usr.email) AND entr.`date` >= emp.valid_from  and entr.`date` <=
emp.valid_until;
```

Image 25 – Executed query to load the FACT_ENTRIES_COMPLIANCE

I was having a problem, there was data missing in the DIMs. Knowing that I had to start debugging DIM by dim searching which Employees, Tasks, Users, Groups were missing. When found the missing data was needed to be inserted in the proper way, with the correct data ranges.

After inserting all the missing records, I re-ran the query, and finally the file records where matching with the FACT_ENTRIES_COMPLIANCE table.

10.3. BI dashboards

Here we arrive to the end of the project. This part consists on representing the DWH into a form that we can understand the metrics and the directors can take decisions.

By now there are only two dashboards created as it requires time. Later more dashboards will be created as we are having lot of information in our DWH.

In the first dashboard we can clearly see the number of hours reported every month distinguishing from the billable ones and the non-billable. Also, in the left of the Dashboard we can see for each role how many billable or non-billable hours are reported. Finally at the bottom, there are the reported hours spitted by client.

We can also create general statistics as we see in the right top of the dashboard.

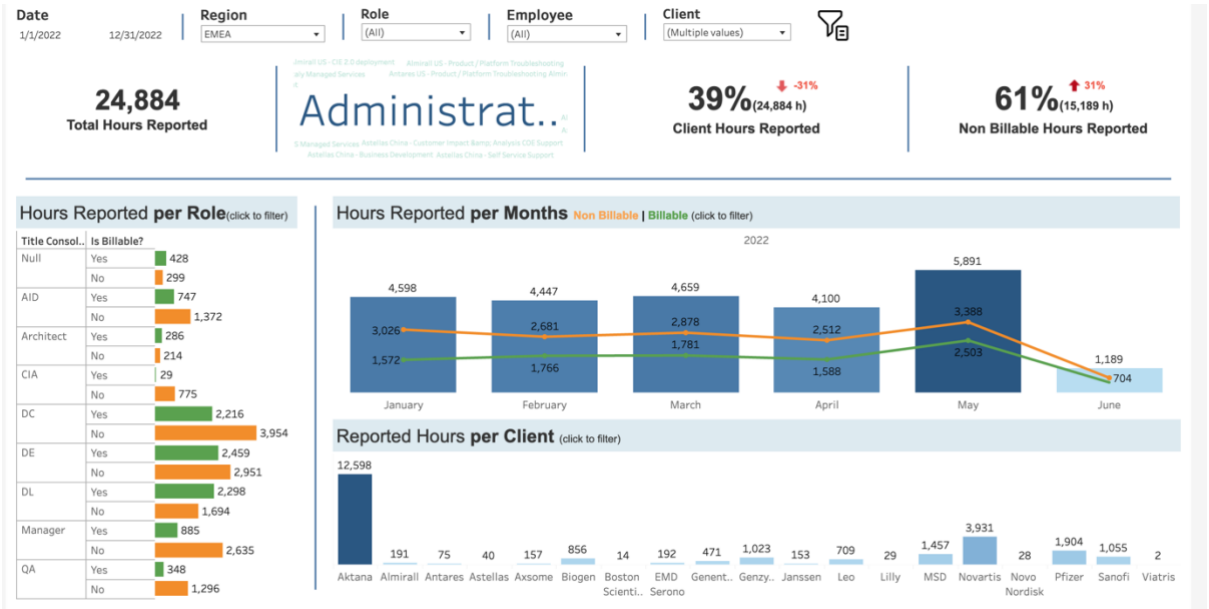


Image 26 – BI dashboard 1

For the second and last dashboard it is represented the hours reported per week. We can also see in the left of the dashboard the hours reported spitted by project and if its billable or non-billable.
Finally, there is the weekly hours report depending on the project and day of the week.

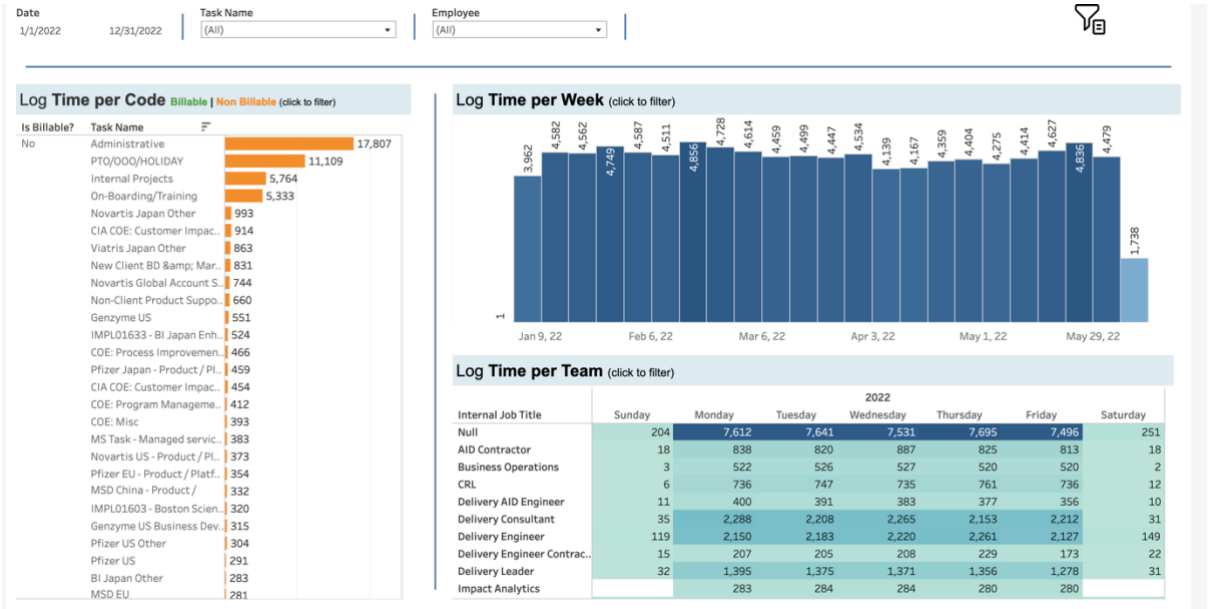


Image 27 – BI dashboard 2

With this dashboard the directors can see how the employees are reporting the hours, see if they are doing it and improve the projects assignation and management.

After finishing this part, the project is concluded. Now it's time to apply it to the company operations team and wait for him to take better decisions as they now have the help of large amount of data.

11. Conclusions

11.1. Objective achievement

Once the work is completed, it is important to look back, get the job done, and reflect about how things were done, why, and whether there were things to improve could have done better.

Valuing the project development form the begin to the end, can be verified if the project has been successfully achieved but could not be finished for lack of time.

Let's analyse the main objectives one by one:

1. Enhance the company's performance.
   We still can not have an answer as it is a big company and we will see the project repercussion in a long therm.

2. Reduce the Operation team time consumption in the process of taking decisions.
   This part was have automatized all the system and now the director doesn't need to run the ETL every week and also doesn't need to update the compliance sheet which was the most time spending.

3. Develop all the workflows to be able to make all the amounts and the filtering of the data that the company's employees report in Timecamp [2].
   All the workflows where developed successfully without giving errors and being capable to be executed automatically every Monday.

4. Develop all the part of the databases which implies the creation of the tables and the triggers to fill the historical tables.
   Also successfully achieved, as you can see in the annex from 2 to 7 all the table triggers are implemented and working successfully, they are all populating the historical tables where there, we have stored all the years information to not lose if the TimeCamp application closes.

5. Make a web-app so that the executives put the changes made in a clearer way.
   This was the only objective that was not possible to be implemented for lack of time since developing a webpage involves a big-time consumption and will remain to implement for the future.

We can say the system has been successfully developed, it solves successfully the formulated problems. The directors and delivery lead only need to check the tableau dashboard reports to get the employees metrics and graphs.

11.2. Future work

This section will detail possible future updates that could be made from this project. Some of the points through which to advance are:

- Integrate more applications in the business intelligence.

This is the best part of this project, it is so expandable, for example, you can add Jira making possible to track the tickets assigned to each employee, the projects a employee is assigned to, in how many time the employees are finishing the tickets. Having this information, you can compare each employee performance between the other employees from the same project. If you look at the figure t, we can see a certain case that a BI could generate for all the employees. It is a comparative between two employees. Having this spider graph the directors can clearly see in which areas the employees are better at. Having this knowledge is much easier to take decisions.

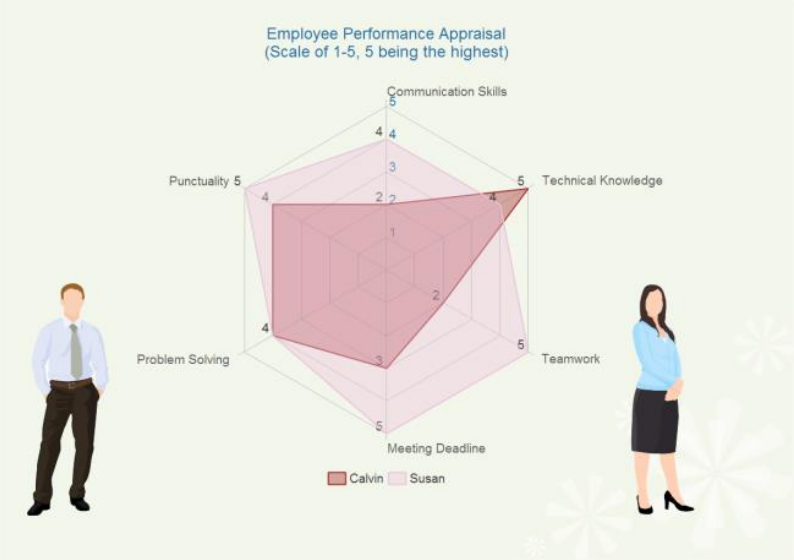


Figure t

There we could also attach salesforce to gather more information. Furthermore, could be added bamboo application to gather even more employee information.

- Generate more Tablau dashboards in order to show more data from our the Data Warehouse

- Develop a web-app.

As I say before, for lack of time, was not possible to end the web-app. It will be a good enhance for our business intelligence as now the director still has to make changes in a smart sheet. By doing the web-app we could make a good graphical interface where would be easier to change the data and, we could limit the changes to be done achieving with this reduce the possibilities to have errors in our system and another time, reducing the director time consumption as it would be optimal.
Personal

11.3. Personal assessment

My personal assessment for this project which I developed as End of Degree has been very positive.

During all this time in the project I have learnt lot of things that will go with me for the rest of my life.

It has forced me to take decisions when was needed to be done. Furthermore, it forced me to solve all the sort of problems I found while developing it. It has shown me how to communicate with the high company executives. It has forced me to improve more my English communication as I had to communicate with people from Americas.

It has allowed me to plan and organize a project from the beginning considering account all the necessary aspects, as if it were a completely real project. And of the in the same way, make a defense of it where the work done is explained.

To conclude, I would like to say the project has helped me on learning to how develop a big project by myself. It made me realize of the large number of setbacks that are possible to have in a project development. It has shown me lots of new technologies unknown for me before this project has started.

12. Bibliografía

[1]
The Data Wharehouse Toolkit: The Definitive Guide to Dimensional Modeling
(Last visited 01-03-2022)

[2]
Timecamp:
https://www.timecamp.com/
(visitado 01-03-2022)

[3]
To know the medium employees salary for each role:
https://www.infoempresa.com/es-es/es/
(visitado 01-03-2022)

[4]
Knime:
https://www.knime.com/
(visitado 01-03-2022)

[5]
DBeaver:
https://dbeaver.io/
(visitado 01-03-2022)

[6]
Kanban:
https://www.atlassian.com/agile/tutorials/how-to-do-kanban-with-jira-software
(visitado 01-03-2022)

[7]
Jira:
https://www.atlassian.com/software/jira
(visitado 01-03-2022)

[8]
Hays:
https://guiasalarial.hays.es/
(visitado 22-03-2022)

[9]
Amazon Web Service:
https://aws.amazon.com/es/rds/mysql/pricing/
(visitado 22-03-2022)

[10]
Gant Desinger:
https://google.gantter.com/gantterforgoogleapps/#amode=normal&fileID=1wec_Y3W2IoHiDk6v3c_n4QB6TceslcHL
(last visited 01-03-2022)

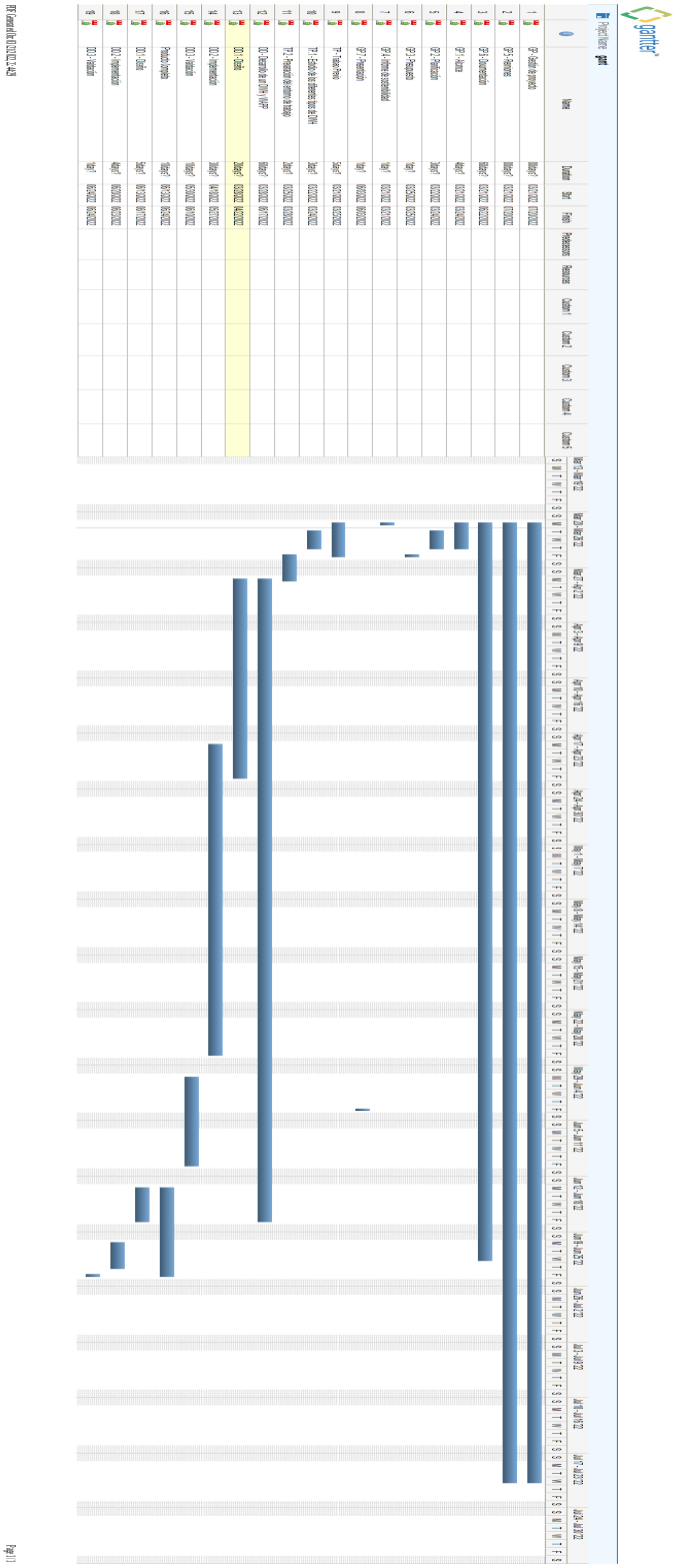
[11]
Knaban EOD:
https://aktana.atlassian.net/jira/software/projects/AP/boards/274
(visitado 01-03-2022)

[12]
Stack Overflow:
https://stackoverflow.com/
(last visited 17-06-2022)
[13]
Python:
https://www.python.org/
(Last visited 13-06-2022)
[14]
Anaconda:
https://www.anaconda.com/
(Last visited 06-06-2022)
[15]
W3Schools repository:
https://www.w3schools.com/
(Last visited 14-06-2022)
[16]
Knime Forum:
https://forum.knime.com/
(Last visited 01-06-2022)
[17]
MySQL for developers:
https://dev.mysql.com/
(Last visited 14-06-2022)
[18]
Youtube Data Warehouse Channel
https://www.youtube.com/c/AlexTheAnalyst
(Last visited 14-06-2022)
[19]
Knime Hub
https://hub.knime.com/
(Last visited 01-06-2022)
[20]
Java Forum
https://www.javatpoint.com/
(Last visited 01-06-2022)
[21]
MySQL information page
https://learnsql.com/
(Last visited 01-06-2022)
[22]
TimeCamp API
https://www.timecamp.com/kb/timecamps-api/
(Last visited 12-04-2022)
[23]
Google Sheets API
https://developers.google.com/sheets/api
(Last visited 13-05-2022)

[24]
Smart sheet API
https://smartsheet-platform.github.io/api-docs/
(Last visited 13-03-2022)
[25]
Excel MySQL connection
https://www.erpsoftwareblog.com/
(Last visited 13-04-2022)
[26]
Bamboo API
https://documentation.bamboohr.com/docs/getting-started
(Last visited 13-04-2022)
[27]
CVS to MySQL
https://www.convertcsv.com/csv-to-sql.htm
(Last visited 13-04-2022)

13. Annex

Anexo 1. Gantt Diagram



Done with Gantter*[10]*

Annex 2

STG_EMPLOYEE Trigger:

```
CREATE DEFINER=`josep.canela`@`%` TRIGGER history_employee
BEFORE UPDATE ON STG_EMPLOYEE FOR EACH ROW
   BEGIN
      IF OLD.hashkey != NEW.hashkey
      THEN
                UPDATE STG_EMPLOYEE_HIST set valid_until=CURRENT_DATE()-
1 where OLD.hashkey = hashkey and valid_until = '2999-12-31 00:00:00';
            INSERT INTO STG_EMPLOYEE_HIST
               (
                  employeeId,
                  first_name,
                  last_name,
                  work_email,
                  hire_date,
                  internal_job_title,
                  job_title,
                  employment_status,
                  termination_date,
                  title_consolidated,
                  region,
                  hashkey,
                  valid_from,
                  valid_until,
                  display_name,
                  preferred_name,
                  mobile,
                  department,
```

```
    location,
    pronouns,
    supervisor,
    FTE
)
VALUES
(
                                NEW.employeeId,
                                NEW.first_name,
                                NEW.last_name,
                                NEW.work_email,
                                NEW.hire_date,
                                NEW.internal_job_title,
                                NEW.job_title,
                                NEW.employment_status,
                                NEW.termination_date,
                                NEW.title_consolidated,
                                NEW.region,
                                NEW.hashkey,
                                CURRENT_DATE(),
                                '2999-12-31',
                                NEW.display_name,
    NEW.preferred_name,
    NEW.mobile,
    NEW.department,
    NEW.location,
    NEW.pronouns,
    NEW.supervisor,
    NEW.FTE
);
```

```
    END IF;
      END
```

Annex 3

STG_TIMECAMP_GROUP trigger

```
CREATE DEFINER=`josep.canela`@`%` TRIGGER history_TimeCampGroup
BEFORE UPDATE ON STG_TIMECAMP_GROUP FOR EACH ROW
   BEGIN
      IF OLD.hashkey != NEW.hashkey
      THEN
                UPDATE STG_TIMECAMP_GROUP_HIST set
valid_until=CURRENT_DATE()-1 where OLD.hashkey = hashkey and valid_until =
'2999-12-31 00:00:00';
            INSERT INTO STG_TIMECAMP_GROUP_HIST
               (
                  group_id,
                                          name,
                                          parent_id,
                                          valid_from,
                                          valid_until,
                                          hashkey
               )
               VALUES
               (
                  NEW.group_id,
                                          NEW.name,
                                          NEW.parent_id,
                                          CURRENT_DATE(),
                                          '2999-12-31',
```

```
                                        NEW.hashkey
            );
    END IF;
      END
```

Annex 4

STG_TIMECAMP_TASK trigger

```
CREATE DEFINER=`josep.canela`@`%` TRIGGER history_TimeCampTask
BEFORE UPDATE ON STG_TIMECAMP_TASK FOR EACH ROW
  BEGIN
    IF OLD.hashkey != NEW.hashkey
    THEN
            UPDATE STG_TIMECAMP_TASK_HIST set
valid_until=CURRENT_DATE()-1 where OLD.hashkey = hashkey and valid_until =
'2999-12-31 00:00:00';
          INSERT INTO STG_TIMECAMP_TASK_HIST
            (
              task_id,
                                        parent_id,
                                        name,
                                        external_task_id,
                                        external_parent_id,
                                        level,
                                        archived,
                                        tags,
                                        budgeted,
                                        budget_unit,
                                        root_group_id,
                                        billable,
```

```
                    note,
                    public_hash,
                    add_date,
                    modify_time,
                    color,
                    hasChildren,
                    assigned_by,
                    valid_from,
                    valid_until,
                    hashkey
)
VALUES
(
    NEW.task_id,
                    NEW.parent_id,
                    NEW.name,
                    NEW.external_task_id,
                    NEW.external_parent_id,
                    NEW.level,
                    NEW.archived,
                    NEW.tags,
                    NEW.budgeted,
                    NEW.budget_unit,
                    NEW.root_group_id,
                    NEW.billable,
                    NEW.note,
                    NEW.public_hash,
                    NEW.add_date,
                    NEW.modify_time,
                    NEW.color,
```

```
                                    NEW.hasChildren,
                                    NEW.assigned_by,

            CURRENT_DATE(),
                '2999-12-31',
            NEW.hashkey
        );
    END IF;
    END
```

Annex 5

STG_TIMECAMP_USER trigger

```
CREATE DEFINER=`josep.canela`@`%` TRIGGER history_TimeCampUser2q
BEFORE UPDATE ON STG_TIMECAMP_USER FOR EACH ROW
   BEGIN
      IF OLD.hashkey != NEW.hashkey
      THEN
               UPDATE STG_TIMECAMP_USER_HIST set
valid_until=CURRENT_DATE()-1 where OLD.hashkey = hashkey and valid_until =
'2999-12-31 00:00:00';
            INSERT INTO STG_TIMECAMP_USER_HIST
               (
                  group_id,
                  user_id,
                  email,
                  display_name,
                  hashkey,
                  valid_from,
                  valid_until
               )
               VALUES
               (
                  NEW.group_id,
                  NEW.user_id,
                  NEW.email,
                  NEW.display_name,
                  NEW.hashkey,
                  CURRENT_DATE(),
                  '2999-12-31'
               );
```

```
    END IF;
  END
```