PROJECT: DOCUFY

DOCUFY

Title: Docufy

Document: Project Plan

Madrid, 14/04/2021

Version 1.0

GENERAL INFORMATION:

Document Type:	Project Plan
Last revised by:	S. Xiao Fernández
Last revision date:	23/04/2021
Approved by	Pablo Almarza
Approval date:	23/04/2021
Team:	N. 5
Status:	Delivered

CHANGE LOG TABLE

VERSION	DATE	CONTENT	PURPOSE
0.1	31/03/2021	Index and structure	Scheduling the document and organising it
0.2	07/04/2021	Requirements catalogue	Having clear the requirements to establish function points and do the estimation
0.3	22/04/2021	Estimation with FP Annex A and B	Completing the document
1.0	23/04/2021	Project Plan finished Meeting minutes added	Revise and correct minimum grammatical or syntactical errors. Observe that everything is complete.

SUMMARY

Docufy is an application that works using data provided by Moodle and lets the students of the Universidad Autónoma de Madrid (UAM) create teams and meetings to complete the assignments proposed by the professors of each different subject. It will also allow the students to schedule meetings in their own teams and consult objectives that will be necessary to complete before the next meeting.

This document is a detailed explanation of the project plan development. It contains the project overview, the estimation of the software system, human resources as well as the time and cost management estimations of this project. It will also show the configuration and process motorization and the test used.

The project estimations were calculated using function points and at the end, we will need 366,15 person/day to develop Docufy and 521,180€. The project is expected to be finished in 405 days.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. Introduction.

This section will introduce information about the Docufy project planning.

1.1 Purpose.

Docufy is an application that will facilitate the students of the UAM to create teams for each assignment of each subject. It will also let the users of this application to plan and create meetings to achieve maximum speed and quality when doing group assignments.

The meetings will be managed in a way a person can see all the scheduled meetings he/she has and the ones that have already passed. The student will be able to see the duration of the call, the objectives to be met to the next meeting, the hour the meeting will or had started, among others.

This document will explain the project plan development. It contains the project overview, the estimation of the software system, human resources as well as the time and cost management estimations of this project. It will also show the configuration and process motorization and the test used.

The project planning is going to be incremental and iterative. We will use a gantt chart for this part, assigning resources needed for each activity. Further explained in section 4. Project management.

1.2 Scope.

The system will have as principal functions:

- Facilitate the work for students to develop their practical assignments and create teams for developing them.
- Facilitate to teachers the evaluation of practical assignments.
- Teachers will have knowledge of the work of each group per team and practical assignment thanks to the statistics subsystem.
- Due to the great number of students from other countries the application will be implemented in spanish and english.
- Only students and teachers registered in Moodle can make use of it.

1.3 Responsibilities.

Project Leader	S. Xiao Fernández Marín
Project Head	Samai García Gonzalez
Project Quality Manager	Carlos García Toledano
Project Documentation Manager	Pablo Almarza Marqués
Customer Representative	Miguel Arnáiz Montes

Table 1. Responsibilities.

1.4 Definitions.

Here we clarify some abbreviations used along the document:

UAM - Universidad Autónoma de Madrid.

Docufy - Name of the project

FP - Function Points

TDI - Total Degree of Influence

AFP - Adjusted Function Points

UFP - Unadjusted Function Points

AF - Adjusted Factor

SEPRO - Software Engineering Project

1.5 Referenced Documentation.

Reference	Title
1	Statements Analysis from the previous delivery
2	Technical Report from the previous delivery
3	SEPRO Unit 4: Software Project Estimation and Planning

Table 2. Referenced documentation.

2. Project Overview.

2.1 Project Description.

The creation of this project is motivated by the need to facilitate students' work and unify in the same system the creation of teams, the development of practical assignments, the creation and hold of meetings...Also it is useful for teachers to correct and evaluate the different assignments they deliver, as they have constance of the meetings held, the time they have spended, and in conclusion the effort they have done in each practical assignment.

2.1.1 Relationship with other systems.

Docufy depends on other systems that provide information about the UAM, about the members of the university, to be exact. It will also need an application to manage the meetings as Docufy is not implementing this function for the moment.

2.1.1.1 Moodle database.

The Moodle database contains information about the UAM professors, subjects, assignments, students, etc.

Docufy needs to be connected with this database as all this information is required to create new groups in the application. With the group information, the application lets the students create teams and after that, Docufy will send the information about the created teams to Moodle, so it can register them in its database.

Docufy will also let the students upload the necessary files, to deliver in a specific assignment, in a special folder called “Practical Assignment Submission”, that will be uploaded to Moodle by default the day of the delivery. We will extend this part in the requirements section.

2.1.1.2 Meeting platform.

A meeting platform is also necessary as Docufy is not implementing this function at the moment.

This platform is going to let the students schedule meetings. When the day and time arrives, Docufy will attach a link to the calendar, where the student can click and join the meeting. He/she can only join the meeting 5 minutes before the time the team proposed arrives.

This platform will also need to let the students share their screens as well as turn on and off their microphones. We will extend this part in the requirements section (2.2 Requirements).

2.1.2 Subsystem decomposition.

Docufy will be divided into six subsystems, this will facilitate a future implementation and modulate the functionalities.

Team management subsystem: Create the team for practical assignments and attach to its members, make notifications, select preferences and view information from the teams.

Meeting scheduling subsystem: See pending or completed teamwork meetings.

Meeting management subsystem: Create the meeting, send reminders, manage the rooms that make up the meeting infrastructure, consult information on meeting development, cancel the meeting.

Meeting-making subsystem: Available tools provided by both the room and the system itself to hold the meeting and work on the internship team, deliver practical assignments.

Meeting tool subsystem: Provides tools for the members of a meeting letting them track their objectives and its progress as well as notifying them when deadlines get close or aren't met.

Statics management subsystem: Provide professors reports in textual and graphical format to see statics about the practical assignments and practical assignments teams and the time and meetings they have held by practical assignment. group and team.

In this way, our subsystem decomposition will look like figure 1:

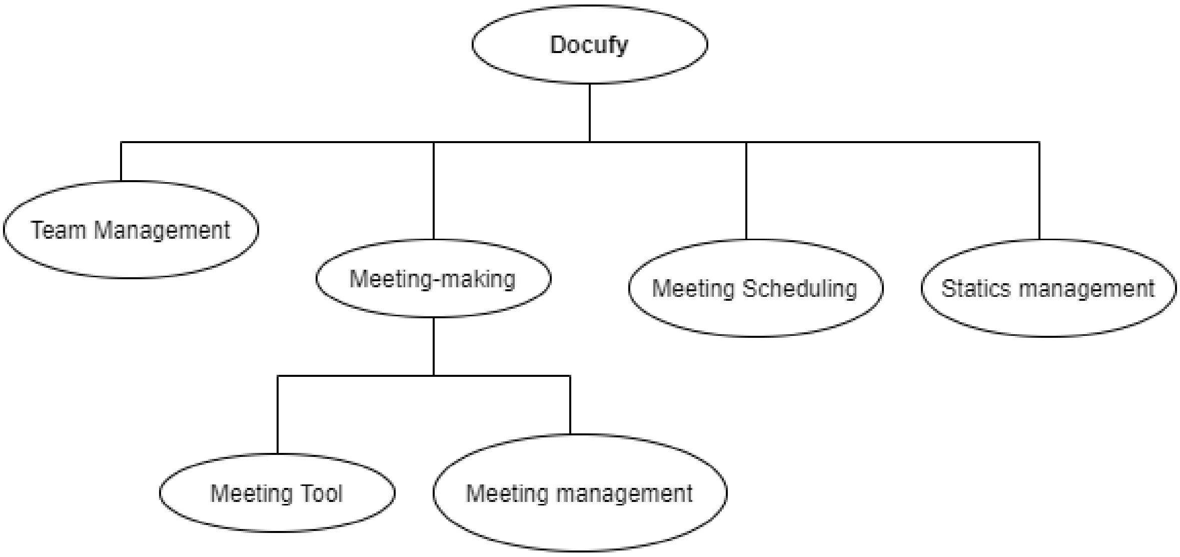


Figure 1. Subsystem decomposition.

2.2 Requirements

2.2.1 Functional Requirements.

Team management subsystem.

FR1: An authenticated user that has logged in previously in Moodle with its credentials (hereinafter, user) without a practical assignment team for a subject who is enrolled can create a team.

1. A file will contain all the information about the team, this is:
    a. Subject name.
    b. Subject with practical assignments.
    c. Group names by subject.
    d. Number of practical assignments with start and end date.
    e. List of participants.
    f. Number of students per group.
    g. Number of students per team.
    h. Student´s name and email.
    i. Subjects in which students are enrolled.
    j. Group to which a student belongs.
    k. Class schedule for each student.
2. Students that remain without a team will be in a file containing:
    a. Student's name.
    b. Student's group.
    c. Subjects in which the student is available.
3. Teams that have been created will be in a file with:
    a. Team's number.
    b. Team’s number of members.
    c. Subject to which the group belongs.

FR2: A team will be created as default for each group: “Help Team”, that will be like a forum.

1. All the students of the same group would be added. This will be provided by Moodle.
    a. List of ids of the students

2. The content of the group consists of:
    a. Question: For students to create new questions or see the ongoing ones

    b. Answers: For students/professors to answer questions

    c. Resolved: After a question has been resolved the label will be changed

    d. Advice: In case a student finds an error on the practice or the theory slides, can warn the other students through this label.

3. A student will be able to post a new question, advice and respond to a question or advice given by other classmates.
    a. Title

    b. Body

FR3: The system will send notifications to the students.

1. Notifications will be sent via email to the students.
    a. Students emails
2. There can be several type of notifications (join request, invitation to a team, team created and team full):
    a. Type of notification
    b. Body of notification

FR4: Every team will be able to modify their preferences and their information.

1. Any petition to change preferences must include (can include “none” in one or two fields):
    a. Days where the team availability is greater.
    b. Estimated duration of the meetings.
    c. Kind of people that are part of the team/are wanted for the team.
2. Any petition to change information must include:
    a. Subject where the team is made.
    b. Description of the team

FR5: Once a team is created in Docufy, it will be registered in the Moodle database.

1. Team information.
    a. Team’s id
    b. Members id
    c. Subject of the team
    d. Description of the team

Meeting scheduling subsystem

FR1: The system will allow team members to consult pending or completed meetings.

1. For the past meetings, the students will be able to consult the information related:
   a. Name
   b. Day, time and duration
   c. Attendees
   d. Time each member has enter the meeting
   e. Achieved tasks in the meeting
   f. Pending tasks
   g. Objectives to be met for the next meeting
2. For pending meetings, students will see:
   a. Name of the meeting
   b. Day, time and duration
   c. Objectives
   d. Topics to be discussed

Meeting management subsystem.

FR1: A user who has logged into moodle and is already on a team can create a meeting.

1. List of users logged in moodle
   a. Id of user
   b. Boolean which tells if the user is logged in or not

2. Users belonging to at least one team
   a. Id of user
   b. List of teams to which the user belongs to

3. Data for creating the meeting
   a. Name of the meeting
   b. Duration of the meeting
   c. Proposed date
   d. Proposed starting hour
   e. User who proposed the meeting

FR2: It can only be carried out by EPS-UAM students.

1. Users of moodle who belong to EPS
   a. Id of user
   b. Faculty to which the user belongs to

FR3: Every team member is able to vote for a date for a specific meeting.

1. System creates poll to vote for the date and hour of the meeting
   a. Days and hours availables
   b. Members of the team

2. System checks every member's calendar to display available dates do vote for in the poll
    a. Id of the team member
    b. Days availables
    c. Time of the day availables
3. A user votes
    a. Id of user who votes
    b. Day voted
    c. Hour voted
4. A user changes its vote
    a. Id of user who votes
    b. Day voted
    c. Hour voted
5. The closes the poll 2 days after it started and sets most voted day, in case of tie the poll is started again
    a. Day and hour voted most
    b. Boolean which represents tie
6. System notifies team members in case that: a date has been agreed, a date has not been agreed
    a. Id of every team member
    b. Text for the notification
7. System updates team members' calendars to include the new meeting if it has been agreed a specific date and hour
    a. Date and hour agreed
    b. Link to the meeting room
    c. Id of every team member

FR4: Meeting management through the streaming platform is carried out by the software system:

1. The app will notify 5 minutes before the start of the meeting.
    a. Id of every team member
    b. Text to notify
    c. Link to the meeting room
2. 5 minutes before starting each meeting, the system opens the room
    a. Petition to external application to open a room in the given link
    b. Link to the meeting room
3. The meeting will end 5 min after all the members have left the meeting.
    a. List of users in the meeting
    b. Petition to external application to close the room in the given link
    c. Link to the meeting room

FR5: Any member of the team can cancel a meeting.

1. Petition of a team member to cancel a meeting
   a. Id of the user
   b. Reason for the cancelation
2. System notifies about the meeting cancelation to the team members
   a. Id of team members
   b. Text of the notification

FR6: The system will send reminders about meetings to be made in two instances: a week before the meeting and the day before the meeting.

1. System notifies a week before a meeting
   a. Id of team members
   b. Text of the notification
2. System notifies one day before the meeting
   a. Id of team members
   b. Text of the notification

Meeting-making subsystem.

FR1: The external application will let the users:

1. Create a new conversation by text
2. Turn on and off their own camera and microphone
3. Record the meeting
4. Share screen
5. Lend the control of their computer

FR2: Docufy will let the students of the same team:

1. Upload and edit a file. The formats supported by Docufy are:
   a. .txt
   b. .doc and .docx
   c. .xlsx
   d. .c, .java, .py, .pl, .asm

2. Upload useful videos explaining things of the assignment to other teammates.
   a. Docufy will show the number of people that have seen the video but not the name of the persons that have seen it.
   b. There will be an option to:
      i. Choose the speed of the video.
      ii. Pause and play.
      iii. Control the video volume.
      iv. Subtitles.
      v. Put in full screen the video.

FR3: When the delivery day of the assignment arrives, Docufy will upload the files to Moodle automatically. The students need to upload the necessary and final files to deliver in a special folder called “Practical Assignment Submission”.

Meeting tool subsystem.

FR1: Docufy will provide a tool for a meeting and its objectives so members can create and assign tasks, set deadlines, mark them as ongoing or completed, edit, etc.

1. First, the system needs a file for each assignment team with their meetings and their objectives
    a. This file has a field with the meeting name.
    b. Each meeting is divided by objectives, so the second field is the meeting objectives’ names.
    c. Then a field stating if that objective has been completed.

2. For each objective there is a file with information about the tasks it is divided in.
    a. Each objective will be divided by tasks, so the first field is the task name.
    b. Then, the student’s name, who is responsible for that task.
    c. Then a deadline for the task.
    d. A field displaying the state of the task (ongoing or completed).

FR2: Docufy will provide a mechanism to know if tasks/deliverables are sent in time.

1. The system needs a file containing the deliverables and their due dates.
    a. The file will have a field with the assignment name.
    b. Another field with the date.

2. To know if a specific team sends a deliverable in time, there is a file containing the teams and their assignments.
    a. There is a field stating the team name.
    b. Another field with the specific assignment name for that team.
    c. And a boolean field stating if it was sent on time, if sent.

Statistics management subsystem.

FR1: The professor can request a report (textual or graphical) of the statistics regarding the meetings taken and grouped by practical assignment and group:

1. Subjects and their practical assignments (each file is a subject).

    a. The first field would be the name of the subject.
    b. The second would be the name of the practical assignment.

2. Practical assignments and statistics about each group (each file is a practical assignment).

    a. First field is the group name.
    b. Then there is a field with the average number of meetings.
    c. Next, a field with the average duration of said meetings.
    d. Then, the percentage of planned meetings that were cancelled.
    e. Lastly, a field with the number of deliverables sent.

FR2: The professor can request a report (textual or graphical) of the statistics regarding the meetings taken by the teams and grouped by practical assignment, group and team:

1. Subjects and their practical assignments (each file is a subject).
    a. The first field would be the name of the subject.
    b. The second would be the name of the practical assignment.

2. Practical assignments and statistics about each group and team (each file is a practical assignment).

    a. First field is the group name.
    b. Second field is the team name.
    c. Then there is a field with the total number of meetings.
    d. Next, a field with the average duration of said meetings.
    e. Then, the percentage of planned meetings that were cancelled.
    f. A field with the number of deliverables sent.
    g. Frequency of meetings during the beginning of the practical assignment.
    h. Frequency of meetings during the middle of the practical assignment.
    i. Frequency of meetings during the end of the practical assignment.

2.2.2 Non-Functional Requirements.

Interface and usability.

NFR1: All the lists that show students, will show at least 6 people by default.

NFR2: There will be an option to change how many people will appear in this list.

NFR3: When a person who is creating a team, chooses another person to join their team, this one will appear blurred to know that he/she is in the list.

NFR4: There should exist a version for smartphones of the interface.

NFR5: The colour of the interface can be changed from blue to green, the same one as Moodle.

NFR6: Each recorded meeting that will be replayed, will open a new window where the meeting will be reproduced.

NFR7: Be able to minimize the meeting without hanging up the call.

NFR8: The meeting will be in a different window than the application.

NFR9: There will be an option to share the screen.

NFR10: Show the number of votes as well as the person who voted. This option is refreshed every 30 minutes.

NFR11: It should be possible to add a colour in order to identify a subject that will take place during the meeting.

NFR12: There will be 2 visible tables. One for having monotonized the task carried out by the team and another one to know if the tasks are met on the date.

NFR13: These tables will be shown during a meeting.

NFR14: The objectives can be displayed in different colours to see the priority.

NFR15: The statistics will be eligible before the generation of the same.

NFR16: Practical assignments, practical assignment groups and teams will be selected before the generation of the statistics.

Documentation requirements.

NFR17: The app will be available in English and Spanish

NFR18: Meetings will save date and duration, therefore it can be ordered by date or duration.

NFR19: All modifications to a delivery, including date and time, will be saved.

NFR20: The tables will follow the APA format.

NFR21: The tables will follow the APA format.

NFR22: The textual reports will be in .txt format.

NFR23: The graphical reports will be in .xlsx format.

Security requirements.

NFR24: User identification consists of the UAM email and a password.

NFR25: The system must encrypt sensitive data transmitted.

NFR26: Only EPS students can access the app

NFR27: Each meeting will only appear for the team it scheduled.

NFR28: The information of the meetings will only be saved on Docufy, not in the external application.

NFR29: The information about the meeting will be saved for 1 year. After this time, it will be erased.

NFR30: A password can be included to enter the meeting.

NFR31: A copy of the delivery will be created.

NFR32: The information from each team is private to the users it concerns, it can't be shown to other users.

NFR33: The professor will have to enter a password to get a report, previously saved when the account is created.

Maintainability and portability requirements.

NFR34: The application will be compatible with Google Chrome, Mozilla Firefox, Safari and Internet Explorer in their latest versions.

NFR35: Meetings can be reproduced in a mobile phone.

NFR36: It should be possible to share screens and talk in the meeting.

NFR37: The time will be updated regularly.

Resources requirements.

NFR38: The app must run smoothly on older machines with say 4 GB of ram and a 5 year old CPU.

NFR39: The app must not occupy more than 15 GB of space.

NFR40: The meetings will be saved until the sum of the sizes is 20 GB, so it will be necessary to delete a recorded meeting in order to have space for another one.

NFR41: The meeting will have a host so the connection depends on the host.

NFR42: The accepted delivery documents will be the ones specified by the teachers.

NFR43: All the tables would be able to be downloaded in a format compatible with Excel.

NFR44: You must be connected to the internet in order to download the statistics.

Performance requirements.

NFR45: The system must be able to handle at least 200 people connected at the same time.

NFR46: It will be possible to fast forward and backward the meeting that has been saved.

NFR47: The meeting must handle the total number of team members.

NFR48: Once the deadline of a delivery is reached, the files uploaded will be sent.

NFR49: The tables must be updated in less than 0.5 seconds after an objective of the meeting is fulfilled or any other relevant action is completed.

NFR50: All modifications will be saved automatically 5 seconds after there aren't more modifications.

NFR51: The download should take less than 5 minutes to download if the sample is large.

Behaviour requirements.

NFR52: The user will be kicked out after being inactive for 30 min.

NFR53: Option to view the calendar in full screen.

NFR54: Change colour of a day depending on the subject of the meeting (the colour can be selected manually)

NFR55: A meeting will take place only if the host connection is at least 5 MB download.

NFR56: The updating of the tables must be in real time, so as soon as a task is completed, the table is updated.

NFR57: Once the report is downloaded, it will appear on the "Downloads" folder of the user's PC.

Availability requirements.

NFR58: There should be a connection between Docufy calendar and a calendar app on the student's mobile so he can see the meetings programmed.

NFR59: Joining a meeting should be possible in every OS and in your mobile phone (in the last case you cannot be the host).

NFR60: Before the deadline, the files can be changed at any time.

NFR61: The tables can be seen while someone is modifying them, and there is no need to modify them if you are just watching.

NFR62: The download cannot be made if there is a meeting transmitting in the sample selected.

Support requirements.

NFR63: There will be a back up with the teams to which the user has belonged.

NFR64: Help button to indicate what should be written in the fields to put the meeting in the schedule.

NFR65: There will be a help button that will answer common questions about the meeting such as why the meeting still runs when everyone is out or why the meeting is not starting.

NFR66: The system will save the previous upload of a delivery, so you can undo a delivery.

NFR67: First time a student is using the table a tutorial will appear to get used to the functionality.

NFR68: Questions can be asked in order that a developer can solve them.

NFR69: There will be a description after the name of the statistics to know what you are getting.

NFR70: There will be an option to get the last statistics you chose.

Verification and reliability.

NFR71: The user's email will be checked with the UAM email database.

NFR72: It will be verified that whoever creates the meeting in the calendar belongs to the UAM.

NFR73: Before joining the meeting, it will be verified that the ones who are joining the meeting belong to the team.

NFR74: One day before the deadline, the system will send a message to all team members so all of them are aware of what is going to be uploaded.

NFR75: The system will ask if you are sure to save the changes in a table before you close it.

NFR76: The system will not give any report if the user does not select any statistic.

2.3 Derivables.

First of all, we will hand to the client a reduced version of the project plan. In addition, after each increment established in the planification of the project we will give to the client the developed code and the documentation generated.

More specifically, after each increment, a specification requirements document will be given to the client, a design document, an executable, a technical manual, and a user manual.

Every document generated will be done in an incremental way, in that form, the deliverables that the client will receive in the last increment will embrace the whole project.

3. Solution to Employ.

Our technical solution consists of a system based on web technology, structured in three layers, client layer, application layer and data layer.

The development of the project will be done in an incremental way, having divided the system in three increments as explained below.

3.1 Software life cycle.

The life cycle that we used while developing this application is the incremental and iterative as said before, using the following phases at each increment: requirements analysis, design, coding, unit tests, integration tests and deployment.
This life cycle is characterized by analyzing the project and dividing it into “subprojects”. These subprojects are the ones that we are going to be developing in each iteration. In each increment we will add more functionality to the system.
It is also characterized by the refinement of each step, and in each iteration it is revised the quality of the software.
In the first increment, we will develop the requirements analysis and a general design with the most important requirements, that we will later code and test.
In the second increment we will improve what we have done in the first one, and complete it with the new requirements or changes the client could have asked us.
The third increment consists in polishing everything developed before, with the purpose of creating the final product just like the client wants it to be.

At the end, the life cycle should look like figure 2:

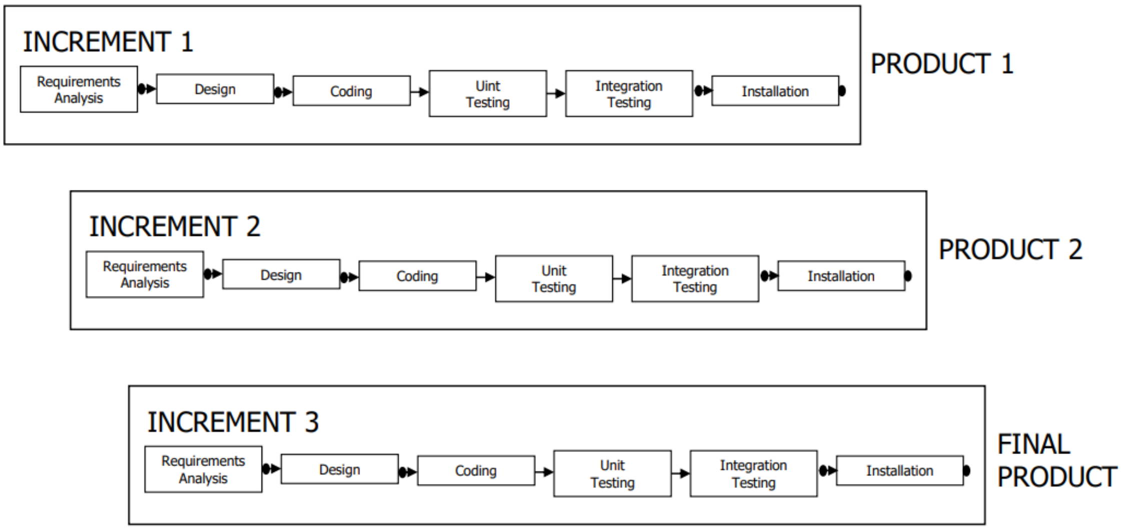


Figure 2. Software life cycle.

3.2 Application interactions.

As said before, the applications that are needed to have iteration with, are principally two. One of them is Moodle, where Docufy takes all the information needed to let the students create teams. The information needed is:

- Subject name
- Subject with practical assignments.
- Group names by subject.
- Number of practical assignments with start and end date.
- List of participants.
- Number of students per group.
- Number of students per team.
- Student´s name and email.
- Subjects in which students are enrolled.
- Group to which a student belongs.
- Class schedule for each student.

Docufy will also need to send information to the Moodle database. This is because when the teams are created, our application will need to register them in the Moodle database.
Docufy needs to have access to Moodle because it will be in charge of uploading the final document into the assignment submission the day of the delivery.

A meeting platform is necessary, it is going to let the students schedule and make meetings. Docufy will attach a link to the calendar, when the day arrives, and the student can click and join the meeting. The student will be allowed to join the meeting 5 minutes before the time the team proposed arrives.

3.3 Architecture.

The architecture used will be a 3-layer model. We can difference:

- Client layer: This layer will be the one that manages the interaction between the client and Docufy. It consists of the PCs, smartphones, tablets and other devices the students use to access the system.

- Application layer: This layer manages the internal logic of Docufy.

- Data layer: This layer is the one that we use as a database. It does not have to be really big as the main data will be on Moodle.

To facilitate understanding we have created a diagram that represents the architecture of the system in a graphical way, figure 3.

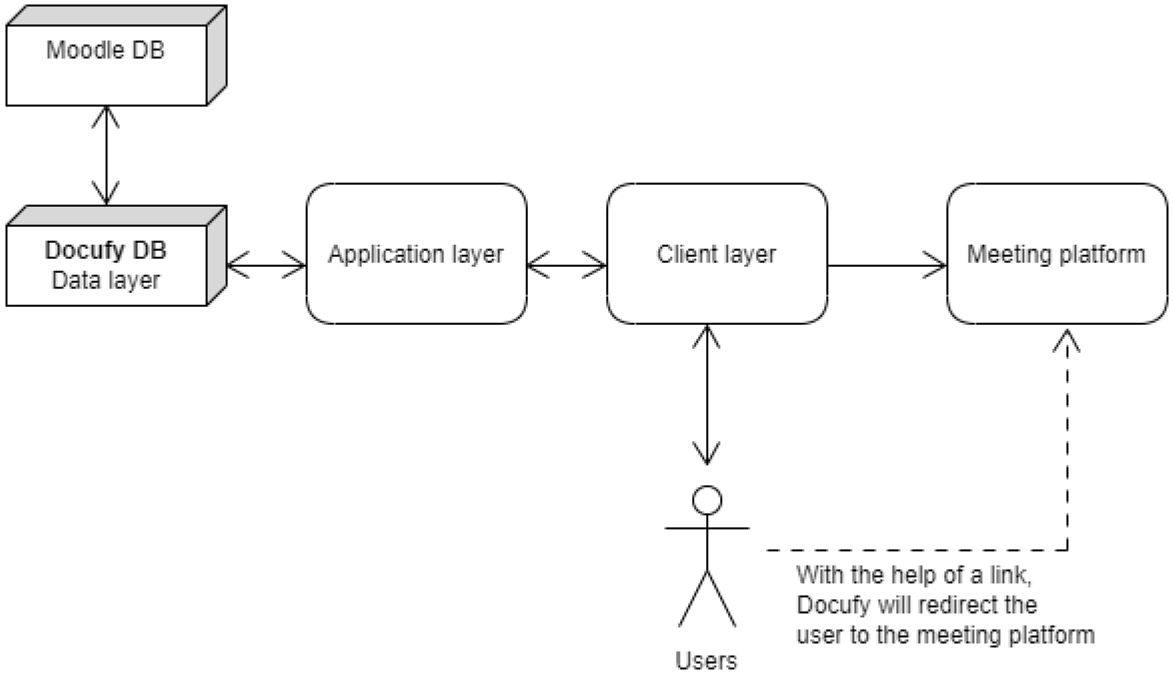

Moodle DB
Docufy DB
Data layer
Application layer
Client layer
Meeting platform
Users
With the help of a link, Docufy will redirect the user to the meeting platform

Figure 3. Architecture.

4. Project Management.

In this section we will present the estimation procedure and the results obtained. Estimating the complexity and size of a project is indispensable for later adjusting as much as possible to the real time. On the contrary, following your own experience and intuition is much less precise than creating a project plan. That's where the importance of project management lies.

4.1 Estimations of the Software System.

For the estimation of the software system time and complexity of Docufy, we have used the method of Function Points. We have used this method in the five subsystems we have (Team management, meeting scheduling, meeting management, meeting tool, statics management). We have divided this section into unadjusted function points and adjusted function points.

4.1.1 Unadjusted function points.

The unadjusted function points are calculated using different parameters such as external inputs (EI), external outputs (EO), external queries (EQ), groups of inner logical data (Internal Logical File, ILF) and groups of external logical data (External Interface Files, EIF).
After analysing the requirements catalogue through the function points, we have obtained the data related to the complexity. Below we have included the tables with the unadjusted function points in each subsystem.

4.1.1.1 Team management.

In table 3 we can see the complexity of the team management subsystem.

	COMPLEXITY						
	Simple	Average	Complex	Simple	Average	Complex	Unadjusted FP (UFP)
Data Functions	Frequency			Weighting			
Internal Logical File (ILF)	7	0	0	7	10	15	49
External Interface File (EIF)	4	0	0	5	7	10	20
Transaction Functions							
External Input (EI)	1	0	1	3	4	6	9
External Output (EO)	2	1	0	4	5	7	13
External Inquiry (EQ)	0	0	0	3	4	6	0
						TOTAL	91,0

Table 3. Team management unadjusted function points.

4.1.1.2 Meeting scheduling.

In table 4 we can see the complexity of the meeting scheduling subsystem.

	COMPLEXITY						
	Simple	Average	Complex	Simple	Average	Complex	Unadjusted FP (UFP)
Data Functions	Frequency			Weighting			
Internal Logical File (ILF)	***0***	***0***	***0***	*7*	*10*	*15*	0
External Interface File (EIF)	***2***	***0***	***0***	*5*	*7*	*10*	10
Transaction Functions							
External Input (EI)	***0***	***0***	***0***	*3*	*4*	*6*	0
External Output (EO)	***0***	***0***	***0***	*4*	*5*	*7*	0
External Inquiry (EQ)	***0***	***1***	***0***	*3*	*4*	*6*	4
						TOTAL	**14,0**

Table 4. Meeting scheduling unadjusted function points.

4.1.1.3 Meeting management

In table 5 we can see the complexity of the meeting management subsystem.

	COMPLEXITY						
	Simple	Average	Complex	Simple	Average	Complex	Unadjusted FP (UFP)
Data Functions	Frequency			Weighting			
Internal Logical File (ILF)	***9***	***0***	***0***	*7*	*10*	*15*	63
External Interface File (EIF)	***9***	***0***	***0***	*5*	*7*	*10*	45
Transaction Functions							
External Input (EI)	***1***	***0***	***3***	*3*	*4*	*6*	21
External Output (EO)	***1***	***0***	***0***	*4*	*5*	*7*	4
External Inquiry (EQ)	***1***	***0***	***0***	*3*	*4*	*6*	3
						TOTAL	**136,0**

Table 5. Meeting scheduling unadjusted function points.

4.1.1.4 Meeting tool.

In table 6 we can see the complexity of the meeting tool subsystem.

	COMPLEXITY						
	Simple	Average	Complex	Simple	Average	Complex	Unadjusted FP (UFP)
Data Functions	Frequency			Weighting			
Internal Logical File (ILF)	1	0	0	7	10	15	7
External Interface File (EIF)	2	0	0	5	7	10	10
Transaction Functions							
External Input (EI)	0	0	0	3	4	6	0
External Output (EO)	1	0	0	4	5	7	4
External Inquiry (EQ)	0	0	0	3	4	6	0
						TOTAL	21,0

Table 6. Meeting tool unadjusted function points.

4.1.1.5 Statics management

In table 7 we can see the complexity of the static management subsystem.

	COMPLEXITY						
	Simple	Average	Complex	Simple	Average	Complex	Unadjusted FP (UFP)
Data Functions	Frequency			Weighting			
Internal Logical File (ILF)	4	0	0	7	10	15	28
External Interface File (EIF)	0	0	0	5	7	10	0
Transaction Functions							
External Input (EI)	0	0	0	3	4	6	0
External Output (EO)	1	1	0	4	5	7	9
External Inquiry (EQ)	0	0	0	3	4	6	0
						TOTAL	37,0

Table 7. Statics management unadjusted function points.

4.1.1.6 Global unadjusted function points (UFP).

As a summary, we have obtained the total unadjusted function points represented in the following table:

Subsystem	Unadjusted FP (UFP)
Team management	91,0
Meeting scheduling	14,0
Meeting Management	136,0
Meeting tool	21,0
Statics management	37,0
TOTAL	299,0

Table 8. Global unadjusted function points.

4.1.2 Adjusted function points.

4.1.2.1 Adjusting Factor

Once we have the unadjusted function points, we calculate the adjustment factor to adjust them. In that way, we have assigned a value to each complexity factor of the general system, as shown in the table 9.

Complexity Factors	TDI
Data Communications	5
Distributed Data Processing	4
Performance	4
Heavily Used Configuration	2
Transaction Rate	0
On-line Data Entry	5
End-User Efficiency	3
On-line Update	4
Complex Processing	0
Reusability	0
Installation Ease	0
Operational Ease	0
Multiple Sites	0
Facilitate Change	0
TOTAL	**27,0**

Table 9. Adjusting factor.

We can see that we have obtained a TDI of 27. The adjusting factor is obtained using the following formula:

AF = (TDI x 0,01) + 0,65 = **0,92**

4.1.2.2 Adjusted function points

Once we have obtained the unadjusted function points and the adjusting factor, now we calculate the adjusted function points of each subsystem using the following formula:

AFP = FP x AF

In this way we obtained the adjusted function points for each subsystem, as we can observe in the table 10.

INCREMENT	SUBSYSTEM	Unadjusted FP	Adjusted FP
INCREMENT 1	**Team management**	91	83,72
	Meeting scheduling	14	12,88
INCREMENT 2	**Meeting Management**	136	125,12
INCREMENT 3	**Meeting tool**	21	19,32
	Statics management	37	34,96

Table 10. Adjusting function points.

4.2 Organizational Structure.

To develop this project plan we have a group of five specialized professionals that cover the needs to develop the project Docufy. Each one of the members of the team is specialized in a different aspect, leading to carry out all the different tasks needed to finish the project.

The team will be composed of:

- A system analyst, that will also be the project leader
- A senior designer
- Two junior designers
- A system technician

The organization of the personnel involved during the project is reflected in the following table:

Activity	Salary
Analyst	400 euros/day
Senior Designer	350 euros/day
Junior Designer 1	200 euros/day
Junior Designer 2	200 euros/day
Technical Staff	300 euros/day

Table 11. Organizational Structure.

Each member of the project will have a journal of 40 working hours a week, 22 days per month.

4.3 Assigned Qualified Personnel.

As we have explained before, the project is divided in three increments, each one made of the phases of Requirements Analysis, Design, Coding, Unit Testing, Integration Testing and Installation.

But there are several points to be taken into account.

The requirements analysis will be performed every time an increment starts, therefore, since we have three increments, three analyses will be done. As the tasks are distributed among the different members of the team, the phases of design, coding and unit testing can be performed in parallel at each increment. These phases should not start before the requirements analysis is finished and revised. Also the implementation for each subsystem should begin after the design has been revised. The integration testing will begin when the unit testing of every increment is finished. The installation phase shall start once the integration tests have been revised.

Increment	Activity Performed	Scheduled Date	
		Start	End
		22/04/21	05/09/22
Increment 1	**Analysis**	22/04/21	28/05/21
	Design	16/06/21	24/09/21
	Coding	19/07/21	26/08/21
	Unit Testing	31/05/21	17/06/21
	Integration Testing	18/06/21	26/07/21
	Installation	27/07/21	13/08/21
Increment 2	**Analysis**	27/09/21	16/11/21
	Design	17/11/21	06/01/22
	Coding	07/01/22	28/02/22
	Unit Testing	01/03/22	24/03/22
	Integration Testing	25/03/22	16/05/22
	Installation	17/05/22	05/09/22
Increment 3	**Analysis**	17/05/22	07/06/22
	Design	08/06/22	29/06/22
	Coding	30/06/22	21/07/22
	Unit Testing	22/07/22	02/08/22
	Integration Testing	03/08/22	24/08/22
	Installation	25/08/22	05/09/22

Table 12. Assigned qualified personnel.

4.4 Time Management.

The system has been divided into five subsystems that will be developed in three increments, following an incremental and iterative life-cycle model.

It is estimated that a single person produces 55,2 *AFPs* during a month, which contains 22 workable days, so we can obtain the effort in the units of person-day. The operation to be done is:

(AFP (Subsystem) * 22 (Workable days)) / Average AFP.

The results are in the figure 13:

INCREMENTS	SUBSYSTEM	Adjusted FP	Effort (person-day)
Increment 1	**Team management**	83,72	33,36
	Meeting scheduling	12,88	5,13
Increment 2	**Meeting Management**	125,12	49,86
Increment 3	**Meeting tool**	19,32	7,7
	Statics management	34,96	13,93

Table 13. Effort person-day.

We can estimate the function points and effort of each increment having the sum of the function points and effort of the subsystems that comprise each increment. We can see that in figure 14:

Increment	Subsystems	Function Points	Effort (person-day)
1	Team management Meeting scheduling	105	124,12
2	Meeting management	136	168,83
3	Meeting tool Statistics management	59	73,2
Total		**300**	**366,15**

Table 14. Function points.

Now we can estimate the time dedicated to each of the phases of the life cycle of each increment, having the effort distribution of each phase. We can visualize that in the Table 15:

INCREMENT	PHASES	% EFFORT	EFFORT (person-day)
1	Requirements Analysis	20%	28
	Design	20%	28
	Coding	20%	28
	Unit Testing	10%	15
	Integration Testing	20%	28
	Installation	10%	14
	Total	100%	136
2	Requirements Analysis	20%	37
	Design	20%	37
	Coding	20%	37
	Unit Testing	10%	18
	Integration Testing	20%	37
	Installation	10%	18
	Total	100%	184
3	Requirements Analysis	20%	16
	Design	20%	16
	Coding	20%	16
	Unit Testing	10%	8
	Integration Testing	20%	16

	Installation	10%	8
	Total	100%	80

Table 15. Phases effort person-day.

For estimating the duration of the project, we have to take into account that the data analyzed before is calculated in person-day. So now we need to do the conversion to working days. We can visualize that in Table 16:

INCREMENT	PHASES	EFFORT	PERSONNEL ASSIGNED	DAYS
1	Requirements Analysis	27,058	3	28
	Design	27,058	3	28
	Coding	27,058	3	28
	Unit Testing	13,529	4	15
	Integration Testing	27,058	3	28
	Installation	13,529	2	14
	Total	135,29	-	**141**
2	Requirements Analysis	36,8	3	37
	Design	36,8	3	37
	Coding	55,2	3	37
	Unit Testing		4	18
	Integration Testing	36,8	3	37
	Installation	18,4	2	18
	Total	184	-	**184**

3	Requirements Analysis	15,69	3	16
	Design	15,69	3	16
	Coding	15,69	3	16
	Unit Testing	7,847	4	8
	Integration Testing	15,69	3	16
	Installation	7,847	2	8
	Total	78,47	-	**80**

Table 16. Days phases.

Although the increments we have are sequential, the project and the personnel assigned will work in a parallel way, which will optimize the time for the project. In Annex C you can consult in detail the planification done.

Four milestones have been defined for each intermediate and final product:

- Milestone at the end of the requirements analysis phase
- Milestone at the end of the design phase
- Milestone at the end of integration tests phases
- Milestone at the end of the installation phase

The MS-Project planning tool has been used for obtaining the project time planning graph or Gantt chart of the project, and the project budget. This will be included in the Annex C.

The estimated time for finishing the project is **405 days**.

4.5 Costs Management.

The elaboration of the system *Docufy* requires a great inversion of hardware, software and human resources. In this section we will estimate the total cost of these resources.

4.5.1 Hardware and Software Resources

The organization has the necessary equipment for the development of the project. Costs and its use are 1,050 euros/month, including both hardware and software.

The total cost for the development equipment that is deduced from the time estimation obtained by means of the Gant diagram, considering an average of 22 working days per month, is:

(€ 1,050 / 22) x (405) days = € 19,330

To develop the project, three workstations must be acquired with a cost of 1,650 euros per station. Another workstation is needed for performance testing with a cost of 3,200 euros.
A new integrated development environment shall also be acquired, at a cost of 1,100 euros per workstation. This environment includes all necessary software for the lifecycle of the project.

Hence, the cost of the material resources that are necessary for the project is:

Material Resources Cost = € 19,330 + € 1,650 x 3 + € 3,200 + € 1,100 = €28,580

4.5.2 Human Resources

The available resources in human terms are the following, and their rates per working day are shown below:

- 1 system analyst, with a cost of 400 euros/day.
- 1 "senior" designer, with a cost of 350 euros/day.
- 2 "junior" designers with a cost of 200 euros/day.
- 1 systems technician, with a cost of 300 euros/day.
- 1 project manager, with a cost of 400 euros/day

From the data presented in section 4.4, we can calculate the working time for each employee:

EMPLOYEE	WORKING DAYS	COST
System analyst	264	105,600€
"Senior" designer	284	99,400€
"Junior" designer	284	56,800
System technician	40	12,000€
Project manager	405	162,000€

Table 17. Human resources.

4.5.3 Project Budget

Finally, a section on Costs Management is included, which evaluates the total estimated budget for the project taking into account the material resources to be used and the qualified personnel to be assigned to the project (Table 18).

RESOURCE	RATE	WORKING DAYS	UNITS	COST
System analyst	400 €/day	264	1	105,600€
“Senior” designer	350€/day	284	1	99,400€
“Junior” designer	200€/day	284	2	113,600€
System technician	300€/day	40	1	12,000€
Project manager	400€/day	405	1	162,000€
Workstation	1,650€	-	3	4,950€
Testing workstation	3,200€	-	1	3,200€
Development environment	1,100€	-	1	1,100€
HW/SW costs	1,050€/month	405	1	19,330€
TOTAL				521,180€

Table 18. Project Budget.

As a conclusion, we have found that the estimated cost for the project is 521,180€. This amount should not be exceeded in order to maintain the expected profits and/or not to incur in losses.

4.6 Quality Management.

Quality = Errors / Function Point.

As we don’t know yet the problems that will arise, we have not created a quality management procedure. But we will continually revise every step taken, and make sure the estimations are fulfilled.

4.7 Risk Management.

Some possible risks about this project are exceeding the profits and budget, or not getting the different increments on time. To avoid possible risks, we have created a Risk Management and Supervision Plan for Docufy, where we identify possible risks, make a risk estimation, risk evaluation, risk management and risk monitoring.

4.8 Acquisitions Management.

The hardware and software elements needed to develop the project are described in section 4.5. We have not decided yet how they will be acquired, this will be decided in later meetings, when the client gives its approval about the budget and we make sure we can start working.

4.9 Documentation Management.

In each different phase, analysis, design and code, documentation is to be created and provided. These documents will be classified by subsystem. The purpose of these documents is to facilitate to the customer the understanding of the decisions taken, possible modifications and additions to the initial requirements. As the application will be intuitive and used by teachers and students, no user manual will be provided.

5. Monitoring and Control.

In this section we will define how the monitoring and control of Docufy will be done. We will include maintainability reunions and tests that we are going to use to check the correct functionality of the project.

5.1 Configuration Management.

The configuration management of Docufy will be the system and software requirements (described in the requirements section), the design documentation about the project and the databases and the verification by the user.

If any configuration of these files changes, the project will need to be reviewed and introduce those new changes to adjust the new management configuration.

These changes will also be quoted in the plan and configuration of the project as well as in the history of versions of the project document.

5.2 Progress Monitorization.

For having continuous progress, the team will meet at least each week to monitor how each phase goes. The reunion will be between the responsible for each phase and the project leader. In table 19 is described how the meeting will be done.

INCREMENT	PHASES	Responsible	Reunion date
1	Requirements Analysis	1	29/05/21
	Design	3	25/09/21
	Coding and Unit Testing	3	18/06/21
	Integration Testing	3	27/07/21
	Installation	2	14/08/21
2	Requirements Analysis	1	17/11/21

	Design	3	07/01/22
	Coding and Unit Testing	3	25/03/22
	Integration Testing	3	17/05/22
	Installation	2	06/09/22
3	Requirements Analysis	1	08/06/22
	Design	3	30/06/22
	Coding and Unit Testing	3	03/08/22
	Integration Testing	3	25/08/22
	Installation	2	06/09/22

Table 19. Progress monotorization.

In the case there is a deviation of the time scheduled for an increment, the efficiency of the team will be improved with the objective of fulfilling the requirements. If the full project is the one with a deviation of time scheduled (not a single increment), there will be a reunion for trying to correct this deviation. If it is not possible to correct it, the delay will be notified to the client.

Besides this, after each increment there will be a reunion with the client to show him how the project is going.

Finally, when the third increment is done, there will be a reunion with all the team and the client. In this reunion the client will test the application.

5.3 Verifications at each Phase.

After the reunions that are specified in the section 5.2, we will verificate the software product or products of each phase. In section 4.2 there are specified these phases and the responsibility of verification each one.

We will make a complete revision of all the product software that is developed at the moment. This verifications are reflected in table 20:

Phase	Product	Responsible
Analysis requirements	DOCUFY-ARQ	Project Quality Manager
Design	DOCUFY-DES	Project Head
Integration tests	DOCUFY-TST	Project Head
Implementation	DOCUFY-UM DOCUFY-TM	Project Head

Table 20. Verification at each phase.

For the analysis requirements, DOCUFY-ARQ, the revision will consist of validating the requirements with the purpose of correcting and completing those requirements, the revision will also look for repeated requirements and inconsistencies.

For the design document, DOCUFY-DES, the verification will check that the design chosen is the one that maximizes the cohesion and minimizes the coupling of the application. This verification will also check the flexibility and reusability of the software.

In the phase of the integration tests, we will make unitary tests and integration tests. Described in the section 5.4 testing and validation.

In the last part, we will develop a technical manual (DOCUFY-TM) and a user manual (DOCUFY-UM). In this phase we will revise these two files, checking that they have the important and relevant information.

5.4 Testing and Validation.

With the objective of finding errors in our project in the analysis, design or codification part, we will make unitary tests for each module. We will use black-box testing every module we have. We will do white-box tests on the modules we think that are the ones carrying most of the work or the ones that are more complex in the project.

After these tests, we will integrate the modules of the new iteration with the ones with the past iterations. We will start with the integration tests. Those that can be of type descendant, ascendant or sandwich. For each set of integrated modules, we will use the type we decide in the design phase of each one.

As the project has to follow non-functional requirements of security, resources, performance, and support, we will also make tests to certify they all work properly.

If all those tests pass and all the increments are done, we will make validation tests of the final product. These tests are carried out by the user and the system analyst. If all the tests pass, all the requirements will be checked.

Finally, the project leader will tell the user to take the acceptance tests. For these tests, the client will check if the software is the one he/she was looking for.

6. Conclusions.

Along with the creation of this document, we have realised the great size of this project, in terms of costs and also of effort and time.

We have observed that due to the magnitude of the project and the effort in person/ day, a great programming team will have to be hired.

On the other hand, although the costs are high, this doesn't mean that the project is not profitable, as this application when finished will save time and resources, which at the end summarizes in saving money.

In conclusion, we think that it is important to have good planning and organisation to make sure that the project objectives and the customer requirements are fulfilled.

ANNEXES

Annex A. Transactional function types

Team management subsystem complexity

Number	Element (Variable names)	Type	RET/FTR	DET	Complexity
1	FR1. An authenticated user that has logged in previously in Moodle with its credentials (hereinafter, user) without a practical assignment team for a subject who is enrolled can create a team.	EI	3	17	*High*
2	ILF1. Information about the team	ILF	1	11	*Low*
3	ILF2. Students without a team	ILF	1	3	*Low*
4	ILF3. Teams already created	ILF	1	3	*Low*
5					
6	FR2. A team will be created as default for each group: “Help Team”, that will be like a forum.	EO	3	7	*Medium*
7	EIF1. All the students of each group are added to the "Help Team"	EIF	1	1	*Low*
8	ILF4. Content of the "Help Team"	ILF	1	4	*Low*
9	EIF2. A student will be able to post a new question, advice and respond to a question or advice given by other classmates.	EIF	1	2	*Low*
10					
11	FR3. The system will send notifications to the students	EO	1	3	*Low*
12	ILF5. Notifications will be sent via email to the student	ILF	1	1	*Low*
13	ILF6. Notification types with their text	ILF	1	2	*Low*
14					
15	FR4. Every team will be able to modify their preferences and their information	EI	1	5	*Low*
16	EIF3. Petition to change team's preferences	EIF	1	3	*Low*
17	EIF4. Petition to change team's information	EIF	1	2	*Low*
18					

19	FR5. Once a team is created in Docufy, it will be registered in the Moodle database	EO	1	4	*Low*
20	ILF7. Teams information	ILF	1	4	*Low*
21					

Table 21. Team management annex A.

INTERNAL LOGICAL FILES (ILF)

Information about the team

DETs

- Subject name.
- Subject with practical assignments.
- Group names by subject.
- Number of practical assignments with start and end date.
- List of participants.
- Number of students per group.
- Number of students per team.
- Student´s name and email.
- Subjects in which students are enrolled.
- Group to which a student belongs.
- Class schedule for each student.

Total: 11 DETs

RETs

- File will contain all the information about the team

Total: 1 RET

CONCLUSION

Complexity is LOW since there are 11 DETs and 1 RET.

Students without a team

DETs

- Student's name.
- Student's group.
- Subjects in which the student is available.

Total: 3 DETs

RETs

- Students that remain without a team

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 3 DETs and 1 RET.

Teams already created

DETs

- Team's number.
- Team’s number of members.
- Subject to which the group belongs.

  Total: 3 DETs

RETs

- Teams that have been created

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 3 DETs and 1 RET.

Content of the "Help Team"

DETs

- Question.
- Answers.
- Resolved.
- Advice.

  Total: 4 DETs

RETs

- Content of the group “Help Team”.

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 4 DETs and 1 RET.

Notifications will be sent via email to the student

DETs

- Students emails

  Total: 1 DET

RETs

- Notifications will be sent via email to the students

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 1 DET and 1 RET.

Notification types with their text

DETs

- Type of notification
- Body of notification
  Total: 2 DETs

RETs

- Type of notifications
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

Teams information

DETs

- Team’s id
- Members id
- Subject of the team
- Description of the team
  Total: 4 DETs

RETs

- Teams information
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 4 DETs and 1 RET.

EXTERNAL INTERFACE FILES (EIF)

All the students of each group are added to the "Help Team"

DETs

- List of ids of the students
  Total: 1 DET

RETs

- Students of the same group
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 1 DET and 1 RET.

A student will be able to post a new question, advice and respond to a question or advice given by other classmates.

DETs

- Title
- Body

  Total: 2 DETs

RETs

- Post a new question

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

Petition to change team's preferences

DETs

- Days where the team availability is greater.
- Estimated duration of the meetings.
- Kind of people that are part of the team/are wanted for the team

  Total: 3 DETs

RETs

- Petition to change preferences

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 3 DETs and 1 RET.

Petition to change team's information

DETs

- Subject where the team is made.
- Description of the team

  Total: 2 DETs

RETs

- Petition to change information

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

EXTERNAL INPUT (EI)

An authenticated user that has logged in previously in Moodle with its credentials (hereinafter, user) without a practical assignment team for a subject who is enrolled can create a team.

DETs

- Subject name.
- Subject with practical assignments.
- Group names by subject.
- Number of practical assignments with start and end date.
- List of participants.
- Number of students per group.
- Number of students per team.
- Student´s name and email.
- Subjects in which students are enrolled.
- Group to which a student belongs.
- Class schedule for each student.
- Student s name.
- Student’s group.
- Subjects in which the student is available.
- Team's number.
- Team’s number of members.
- Subject to which the group belongs.

Total: 17 DETs

RETs

- Information about the team
- Students without a team
- Teams already created

Total: 3 RET

CONCLUSION
Complexity is HIGH since there are 1 DETs and 1 RET.

Every team will be able to modify their preferences and their information

DETs

- Days where the team availability is greater.
- Estimated duration of the meetings.
- Kind of people that are part of the team/are wanted for the team.
- Subject where the team is made.
- Description of the team
  Total: 5 DETs

RETs

- Petition to change preferences
- Petition to change information
  Total: 2 RET

CONCLUSION
Complexity is SIMPLE since there are 5 DETs and 2 RETs.

EXTERNAL OUTPUT (EO)

A team will be created as default for each group: "Help Team", that will be like a forum

DETs

- List of ids of the students
- Question.
- Answers.
- Resolved.
- Advice.
- Title
- Body
  Total: 7 DET

RETs

- Students of the same group
- Content of the group
- Post a new question

  Total: 3 RET

CONCLUSION

Complexity is MEDIUM since there are 7 DETs and 3 RETs.

The system will send notifications to the students

DETs

- List of ids of the students

  Total: 1 DET

RETs

- Students of the same group

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 1 DET and 1 RET.

Once a team is created in Docufy, it will be registered in the Moodle database

DETs

- Team's id
- Members id
- Subject of the team
- Description of the team

  Total: 4 DET

RETs

- Teams information.

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 4 DETs and 1 RET.

Meeting scheduling subsystem complexity

Number	Element (Variable names)	Type	RET/FTR	DET	Complexity
1	FR1. The system will allow team members to consult pending or completed meetings	EQ	2	11	*Medium*
2	EIF1. Past meetings records	EIF	1	7	*Low*
3	EIF2. Pending meetings records	EIF	1	4	*Low*
4					

Table 22. Meeting scheduling annex A.

EXTERNAL INTERFACE FILES (EIF)

Past meetings records

DETs

- Name
- Day, time and duration
- Attendees
- Time each member has enter the meeting
- Achieved tasks in the meeting
- Pending tasks
- Objectives to be met for the next meeting

  Total: 7 DET

RETs

- Information related to past meetings

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 7 DET and 1 RETs.

Pending meetings records

DETs

- Name of the meeting
- Day, time and duration
- Objectives
- Topics to be discussed

  Total: 4 DET

RETs

- Information related to pending meetings
  Total: 1 RET

CONCLUSION
Complexity is LOW since there are 4 DET and 1 RET.

EXTERNAL INQUIRY (EQ)

The system will allow team members to consult pending or completed meetings

DETs

- Name
- Day, time and duration
- Attendees
- Time each member has enter the meeting
- Achieved tasks in the meeting
- Pending tasks
- Objectives to be met for the next meeting
- Name of the meeting
- Day, time and duration
- Objectives
- Topics to be discussed
  Total: 11 DET

RETs

- Information related to past meetings
- Information related to pending meetings
  Total: 2 RET

CONCLUSION
Complexity is MEDIUM since there are 11 DETs and 2 RETs.

Meeting management subsystem complexity

Number	Element (Variable names)	Type	RET/FTR	DET	Complexity
1	FR1. A user who has logged into moodle and is already on a team can create a meeting	EI	3	8	*High*
2	EIF1. Users logged in moodle	EIF	1	2	*Low*
3	ILF1. Users belonging to a team	ILF	1	2	*Low*
4	EIF2. Data for creating meeting	EIF	1	5	*Low*
5					
6	FR2. It can only be carried out by EPS-UAM students	EQ	1	2	*Low*
7	EIF3. Users of moodle who belong to EPS	EIF	1	2	*Low*
8					
9	FR3.Every team member is able to vote for a date for a specific meeting	EI	7	19	*High*
10	ILF2. The system creates a poll to vote for the date and hour of the meeting	ILF	1	3	*Low*
11	EIF4. A user votes for an specific date and hour	EIF	1	3	*Low*
12	EIF5. A user changes his vote	EIF	1	3	*Low*
13	ILF3. The closes the poll 2 days after it started and sets most voted day, in case of tie the poll is started again	ILF	1	2	*Low*
14	EIF6. System checks every member's calendar to display available dates to vote for in the poll	EIF	1	3	*Low*
15	IFL6. System notifies team members in case that: a date has been agreed, a date has not been agreed	ILF	1	2	*Low*
16	IFL7. System updates team members' calendars if a date has been agreed for the meeting	ILF	1	3	*Low*
17					
18	FR4. Meeting management through the streaming platform is carried out by the software system	EI	3	8	*High*
19	ILF8. Send notification 5 minutes before the meeting	ILF	1	3	*Low*
20	EIF7. Meeting room opens 5 minutes before the meeting starts	EIF	1	2	*Low*
21	EIF8. Meeting room closes 5 minutes after every member has left the meeting room	EIF	1	3	*Low*
22					
23	FR5. Any member of the team can cancel a meeting	EI	2	4	*Low*
24	ILF9. System notifies team members about the canceled meeting	ILF	1	2	*Low*
25	EIF9. Petition of a team member to cancel the meeting	EIF	1	2	*Low*
26					
27	FR6: The system will send reminders about meetings to be made in two instances:	EO	2	4	*Low*

	a week before the meeting and the day before the meeting				
28	ILF10. System notifies a week before a meeting	ILF	1	2	*Low*
29	ILF13. System notifies one day before the meeting	ILF	1	2	*Low*
30					

Table 23. Meeting management annex A.

INTERNAL LOGICAL FILES (ILF)

Users belonging to a team

DETs

- Id of user

- List of teams to which the user belongs to
  Total: 2 DETs

RETs

- List of users that belong to at least 1 team
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

The system creates a poll to vote for the date and hour of the meeting

DETs

- Days and hours availables

- Members of the team
  Total: 2 DETs

RETs

- Information about the availability and participants of the vote
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

The system closes the poll 2 days after it started and sets most voted day, in case of tie the poll is started again

DETs

- Day and hour voted most

- Boolean which represents tie
  Total: 2 DETs

RETs

- Information about the results of the poll
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

System notifies team members in case that: a date has been agreed, a date has not been agreed

DETs

- Id of every team member
- Text for the notification

Total: 2 DETs

RETs

- Information about the notification to be sent about the creation of a meeting

Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

System updates team members' calendars if a date has been agreed for the meeting

DETs

- Date and hour agreed
- Link to the meeting room
- Id of every team member

Total: 3 DETs

RETs

- Information about the new date setted for the meeting

Total: 1 RET

CONCLUSION

Complexity is LOW since there are 3 DETs and 1 RET.

Send notification 5 minutes before the meeting

DETs

- Id of every team member
- Text to notify
- Link to the meeting room

Total: 3 DETs

RETs

- Information about the notification to be sent before the meeting
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 3 DETs and 1 RET.

System notifies team members about the canceled meeting

DETs

- Id of team members
- Text of the notification
  Total: 2 DETs

RETs

- Information about the notification about a cancelled meeting
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

System notifies a week before a meeting

DETs

- Id of team members
- Text of the notification
  Total: 2 DETs

RETs

- Information about the notification about a future meeting
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

System notifies one day before the meeting

DETs

- Id of team members
- Text of the notification
  Total: 2 DETs

RETs

- Information about the notification about a future meeting
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

EXTERNAL INTERFACE FILES (EIF)

Users logged in moodle

DETs

- Id of user
- Boolean which tells if the user is logged in or not

  Total: 2 DET

RETs

- Information about the users currently logged in moodle

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DET and 1 RET.

Data for creating meeting

DETs

- Name of the meeting
- Duration of the meeting
- Proposed date
- Proposed starting hour
- User who proposed the meeting

  Total: 5 DET

RETs

- Information about the data needed for creating a meeting

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 5 DET and 1 RET.

Users of moodle who belong to EPS

DETs

- Id of user
- Faculty to which the user belongs to

  Total: 2 DET

RETs

- Information about the data needed for creating a meeting

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DET and 1 RET.

A user votes for an specific date and hour

DETs

- Id of user who votes

- Day voted
- Hour voted

Total: 3 DET

RETs

- Information about the user and its vote

Total: 1 RET

CONCLUSION

Complexity is LOW since there are 3 DET and 1 RET.

A user changes his vote

DETs

- Id of user who votes
- Day voted
- Hour voted

Total: 3 DET

RETs

- Information about the user and its new vote

Total: 1 RET

CONCLUSION

Complexity is LOW since there are 3 DET and 1 RET.

System checks every member's calendar to display available dates to vote for in the poll

DETs

- User id
- Date available
- Available for all boolean

Total: 3 DET

RETs

- Information about the available dates of all the members

Total: 1 RET

CONCLUSION

Complexity is LOW since there are 3 DET and 1 RET.

Meeting room opens 5 minutes before the meeting starts

DETs

- Petition to external application to open a room in the given link
- Link to the meeting room

Total: 2 DET

RETs

- Information about the available dates of all the members

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DET and 1 RET.

Meeting room closes 5 minutes after every member has left the meeting room

DETs

- List of users in the meeting
- Petition to external application to close the room in the given link
- Link to the meeting room

  Total: 3 DET

RETs

- Information about the available dates of all the members

  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 3 DET and 1 RET.

EXTERNAL INPUT (EI)

A user who has logged into moodle and is already on a team can create a meeting

DETs

- Id of user
- Boolean which tells if the user is logged in or not
- Name of the meeting
- Duration of the meeting
- Proposed date
- Proposed starting hour
- User who proposed the meeting
- Id of user
- List of teams to which the user belongs to

  Total: 9 DET

RETs

- List of users logged in moodle
- Users belonging to at least one team
- Data for creating the meeting

  Total: 3 RET

CONCLUSION

Complexity is HIGH since there are 9 DET and 3 RET.

Every team member is able to vote for a date for a specific meeting

DETs

- Id of every team member
- Text to notify
- Link to the meeting room
- Petition to external application to open a room in the given link
- Link to the meeting room
- List of users in the meeting
- Petition to external application to close the room in the given link
- Link to the meeting room

Total: 8 DET

RETs

- The app will notify 5 minutes before the start of the meeting
- 5 minutes before starting each meeting, the system opens the room
- The meeting will end 5 min after all the members have left the meeting.

Total: 3 RET

CONCLUSION

Complexity is HIGH since there are 8 DET and 3 RET.

Any member of the team can cancel a meeting

DETs

- Id of the user
- Reason for the cancelation
- Id of team members
- Text of the notification

Total: 4 DET

RETs

- Petition of a team member to cancel a meeting
- System notifies about the meeting cancelation to the team members

Total: 2 RET

CONCLUSION

Complexity is LOW since there are 4 DET and 2 RET.

EXTERNAL OUTPUT (EO)

The system will send reminders about meetings to be made in two instances:

DETs

- Id of team members
- Text of the notification

- Id of team members
- Text of the notification

Total: 4 DET

RETs

- System notifies a week before a meeting
- System notifies one day before the meeting

Total: 2 RET

CONCLUSION

Complexity is LOW since there are 4 DETs and 2 RETs.

EXTERNAL INQUIRY (EQ)

It can only be carried out by EPS-UAM students

DETs

- Id of user
- Boolean which tells if the user is logged in or not
- Id of user
- List of teams to which the user belongs to
- Name of the meeting
- Duration of the meeting
- Proposed date
- Proposed starting hour
- User who proposed the meeting

Total: 9 DET

RETs

- List of users logged in moodle
- Users belonging to at least one team
- Data for creating the meeting

Total: 3 RET

CONCLUSION

Complexity is LOW since there are 9 DETs and 3 RETs.

Meeting tool subsystem complexity

Number	Element (Variable names)	Type	RET/FTR	DET	Complexity
1	FR1: Tasks/Objectives completion tracking	EI	2	7	*Medium*
2	FR1: Tasks/Objectives completion tracking	EO	2	7	*Medium*
3					
4	ILF1: Objectives of the meeting	ILF	1	3	*Low*
5	EIF1: Information about the objectives' tasks	EIF	1	4	*Low*
6					
7					
8					
9	FR2: Deliverables sent tracking	EO	2	5	*Low*
10					
11	ILF2: Deadline tracking for deliverables	ILF	1	2	*Low*
12	EIF2: Deliverables and their deadlines	EIF	1	3	*Low*
13					

Table 24. Meeting tool annex A.

INTERNAL LOGICAL FILES (ILF)

Objectives of the meeting

DETs

- This file has a field with the meeting name.
- Each meeting is divided by objectives, so the second field is the meeting objectives' names.
- Then a field stating if that objective has been completed.
  Total: 3 DETs

RETs

- File for each assignment team with their meetings and their objectives
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 3 DETs and 1 RET.

Deadline tracking for deliverables

DETs

- The file will have a field with the assignment name.
- Another field with the date.
  Total: 2 DETs

RETs

- File containing the deliverables and their due dates

Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

EXTERNAL INTERFACE FILES (EIF)

Information about the objectives' tasks

DETs

- Each objective will be divided by tasks, so the first field is the task name.
- Then, the student's name who is responsible for that task.
- Then a deadline for the task.
- A field displaying the state of the task (ongoing or completed).

Total: 4 DET

RETs

- File with information about the tasks it is divided

Total: 1 RET

CONCLUSION

Complexity is LOW since there are 4 DET and 1 RET.

Deliverables and their deadlines

DETs

- There is a field stating the team name.
- Another field with the specific assignment name for that team.
- And a boolean field stating if it was sent on time, if sent.

Total: DET

RETs

-

Total: RET

CONCLUSION

Complexity is LOW since there are DET and RET.

EXTERNAL INPUT (EI)

Tasks/Objectives completion tracking

DETs

- This file has a field with the meeting name.
- Each meeting is divided by objectives, so the second field is the meeting objectives' names.
- Then a field stating if that objective has been completed.
- Each objective will be divided by tasks, so the first field is the task name.
- Then, the student's name who is responsible for that task.

- Then a deadline for the task.
- A field displaying the state of the task (ongoing or completed).
  Total: 7 DETs

RETs

- Objectives of the meeting
- Information about the objectives' tasks
  Total: 2 RETs

CONCLUSION

Complexity is MEDIUM since there are 7 DETs and 2 RETs.

EXTERNAL OUTPUT (EO)

Tasks/Objectives completion tracking

DETs

- This file has a field with the meeting name.
- Each meeting is divided by objectives, so the second field is the meeting objectives' names.
- Then a field stating if that objective has been completed.
- Each objective will be divided by tasks, so the first field is the task name.
- Then, the student's name who is responsible for that task.
- Then a deadline for the task.
- A field displaying the state of the task (ongoing or completed).
  Total: 7 DETs

RETs

- Objectives of the meeting
- Information about the objectives' tasks
  Total: 2 RETs

CONCLUSION

Complexity is MEDIUM since there are 7 DETs and 2 RETs.

Deliverables sent tracking

DETs

- The file will have a field with the assignment name.
- Another field with the date.
- There is a field stating the team name.
- Another field with the specific assignment name for that team.

- And a boolean field stating if it was sent on time, if sent.
  Total: 5 DET

RETs

- Deadline tracking for deliverables
- Deliverables and their deadlines
  Total: 2 RET

CONCLUSION

Complexity is LOW since there are 5 DETs and 2 RETs.

Statistics Management subsystem complexity

Number	Element (Variable names)	Type	RET/FTR	DET	Complexity
1	FR1: Statistics by assignment and group	EO	2	7	*Medium*
2	ILF1: Subjects and their practical assignments	ILF	1	2	*Low*
3	ILF2: Statistics by group and practical assignment	ILF	1	5	*Low*
4					
5					
6	FR2: Statistics by assignment, group and team	EO	2	11	*Medium*
7	ILF1: Subjects and their practical assignments	ILF	1	2	*Low*
8	ILF2: Statistics by group, practical assignment and team	ILF	1	9	*Low*
9					

Table 25. Statics management annex A.

INTERNAL LOGICAL FILES (ILF)

Subjects and their practical assignments

DETs

- The first field would be the name of the subject.
- The second would be the name of the practical assignment.
  Total: 2 DETs

RETs

- Subjects and their practical assignments
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

Statistics by group and practical assignment

DETs

- First field is the group name.
- Then there is a field with the average number of meetings.
- Next, a field with the average duration of said meetings.
- Then, the percentage of planned meetings that were cancelled.
- Lastly, a field with the number of deliverables sent.
  Total: 5 DETs

RETs

- Practical assignments and statistics about each group
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 5 DETs and 1 RET.

Subjects and their practical assignments

DETs

- The first field would be the name of the subject.
- The second would be the name of the practical assignment.
  Total: 2 DETs

RETs

- Subjects and their practical assignments
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

Statistics by group, practical assignment and team

DETs

- First field is the group name.
- Second field is the team name.
- Then there is a field with the total number of meetings.
- Next, a field with the average duration of said meetings.
- Then, the percentage of planned meetings that were cancelled.
- A field with the number of deliverables sent.
- Frequency of meetings during the beginning of the practical assignment.

- Frequency of meetings during the middle of the practical assignment.
- Frequency of meetings during the end of the practical assignment.
  Total: 2 DETs

RETs

- Practical assignments and statistics about each group and team
  Total: 1 RET

CONCLUSION

Complexity is LOW since there are 2 DETs and 1 RET.

EXTERNAL OUTPUT (EO)

The professor can request a report (textual or graphical) of the statistics regarding the meetings taken and grouped by practical assignment and group.

DETs

- The first field would be the name of the subject.
- The second would be the name of the practical assignment.
- First field is the group name.
- Then there is a field with the average number of meetings.
- Next, a field with the average duration of said meetings.
- Then, the percentage of planned meetings that were cancelled.
- Lastly, a field with the number of deliverables sent.
  Total: 7 DET

RETs

- Subjects and their practical assignments
  Total: 2 RET

CONCLUSION

Complexity is MEDIUM since there are 7 DETs and 2 RETs.

The professor can request a report (textual or graphical) of the statistics regarding the meetings taken by the teams and grouped by practical assignment, group and team

DETs

- The first field would be the name of the subject.
- The second would be the name of the practical assignment.

- First field is the group name.
- Second field is the team name.
- Then there is a field with the total number of meetings.
- Next, a field with the average duration of said meetings.
- Then, the percentage of planned meetings that were cancelled.
- A field with the number of deliverables sent.
- Frequency of meetings during the beginning of the practical assignment.
- Frequency of meetings during the middle of the practical assignment.
- Frequency of meetings during the end of the practical assignment.
  Total: 11 DETs

RETs

- Subjects and their practical assignments
- Practical assignments and statistics about each group and team
  Total: 2 RETs

CONCLUSION

Complexity is MEDIUM since there are 11 DETs and 2 RETs.

Annex B. Adjustment factor

In this annex we will justify how we value the different characteristics to get the complexity factor value.

Data Communications: As the application should be developed to be used by different devices and from different platforms, that means that more than one front-end computer should be used and the application supports more than one communications protocol. That's why we have given this feature a value of 5.

Distributed Data Processing: As the data inputs, updates and queries are done in an online way and in both directions, and the system has to communicate with external software systems, we have attributed this feature a value of 4.

Performance: As the system shall carry out high demand operations, such as creation of teams for practical assignments and/or meetings scheduling and creation in the shortest possible time to be defined by the team, a special design is required, so this attribute has a value of 4.

Heavily Used Configuration: Although some operations should be done in the shortest time possible, this does not require a special effort, that's why we have given this feature a value of 2.

Transaction Rate: As there is no specification about this point, it's value is 1.

On-line Data Entry: As every input is done in an online way, and usability mechanisms such as feedback, simplified help and language selection shall be provided, we can conclude that more than 30% of the inputs are interactive, and the value is 5.

End-User Efficiency: The final application will include the following functionalities for the user: creation of teams for practical assignments and/or meetings scheduling and creation..etc. This, combined with the customizable and adaptable design, suppose a punctuation of 3, as there are no specific user efficiency requirements.

On-line Update: As one of the functionalities of this application is to store data about schedules, meetings, assignments, and in order to avoid plagiarism between students, the protection against data loss is essential and it has been especially designed and programmed in the system, so this will have a value of 4.

Complex Processing: As the application doesn´t have complex processing and doesn't make use of other applications for any process, this has a value of 0.

Reusability: They didn’t ask us about the code to be reusable, so we deduce it doesn't have reusability: 0.

Installation Ease: As the application requires no special considerations for its installation, the installation ease will be 0.

Operational Ease: As no special operation needs have been defined by the user, we give this feature a value of 0.

Multiple Sites: In principle, there are no requirements to consider the need to install the application in different locations, so we suppose a punctuation of 0.

Facilitate Change: There is no user requirement to facilitate change (for example different system inputs, flexibility of queries or output, etc.), so we give this feature a value of 0.

Annex C. Gantt Chart

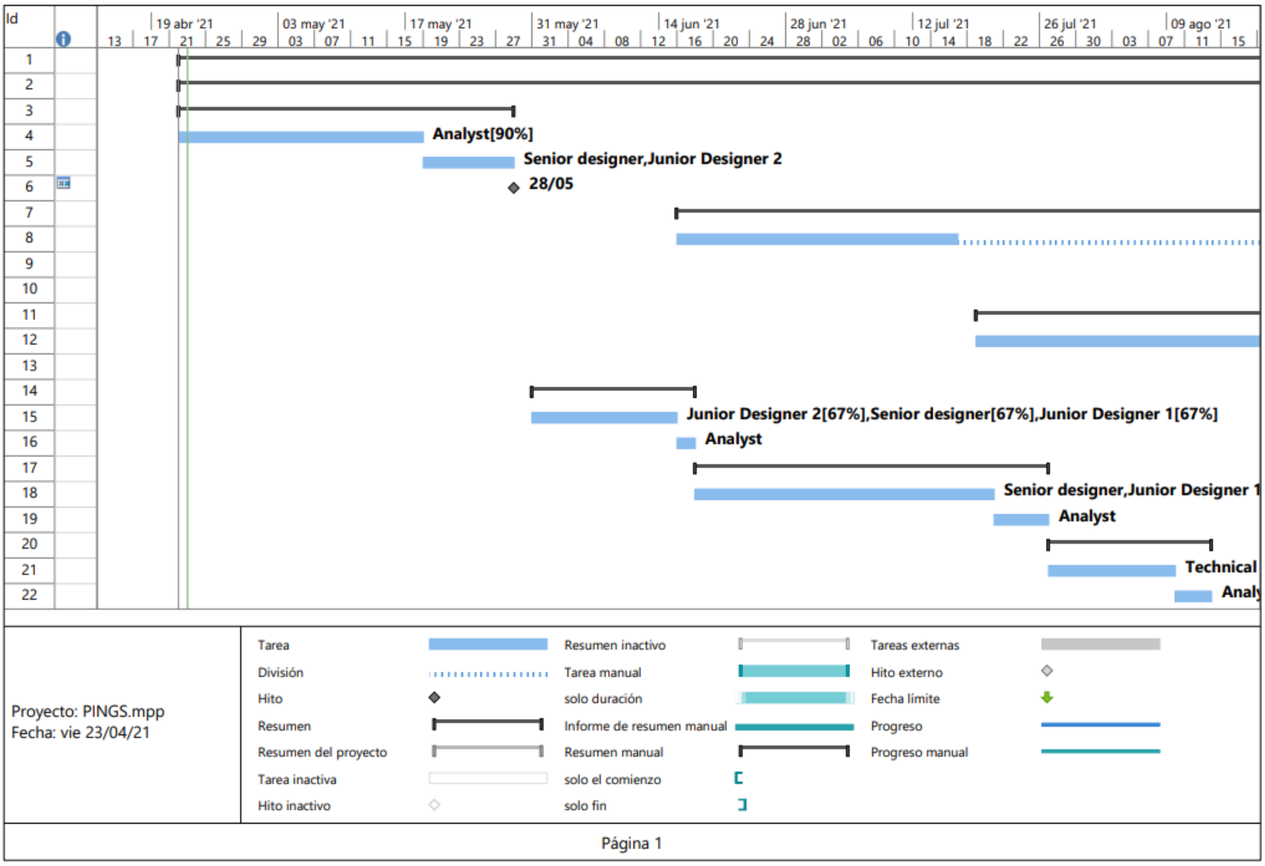


Figure 4. Gantt chart 1. First part of the first increment.

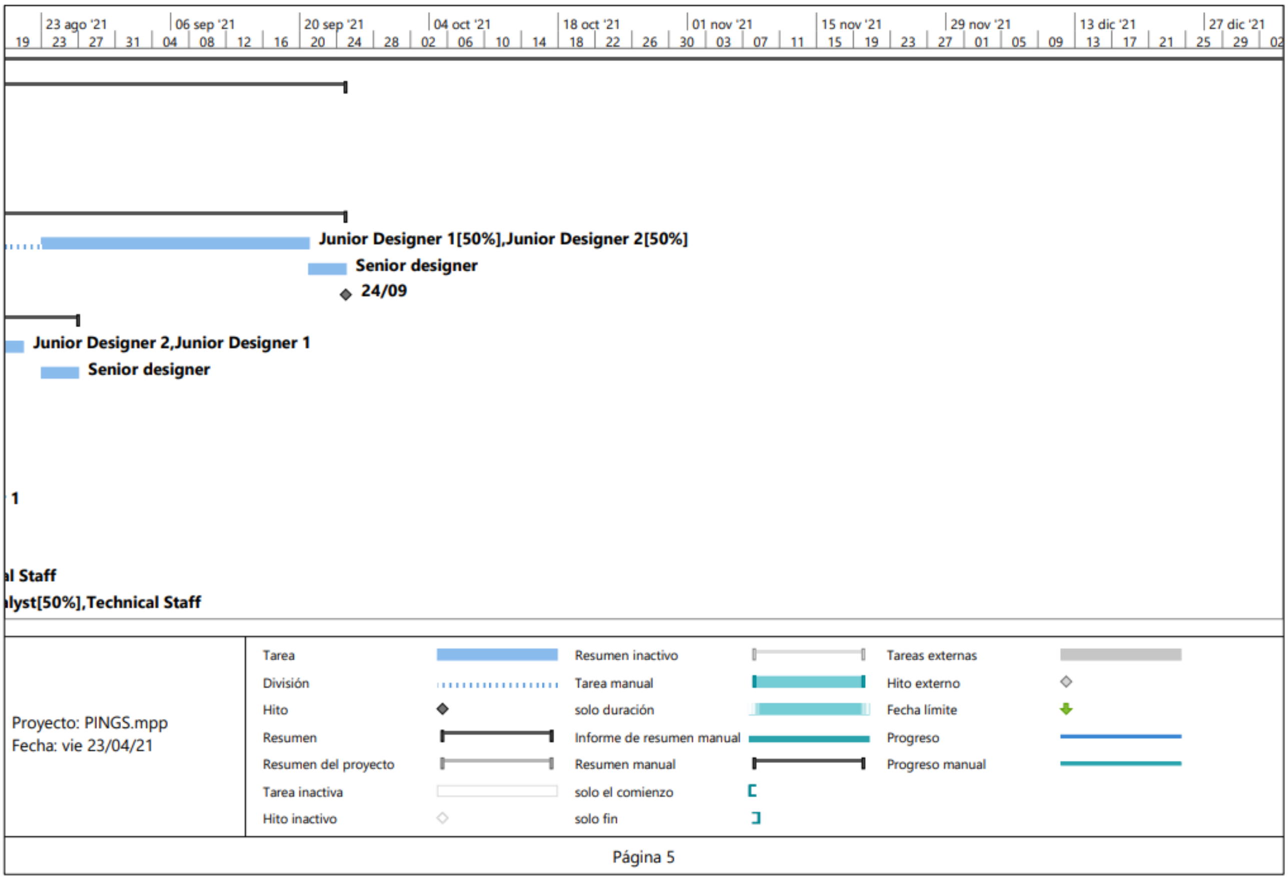
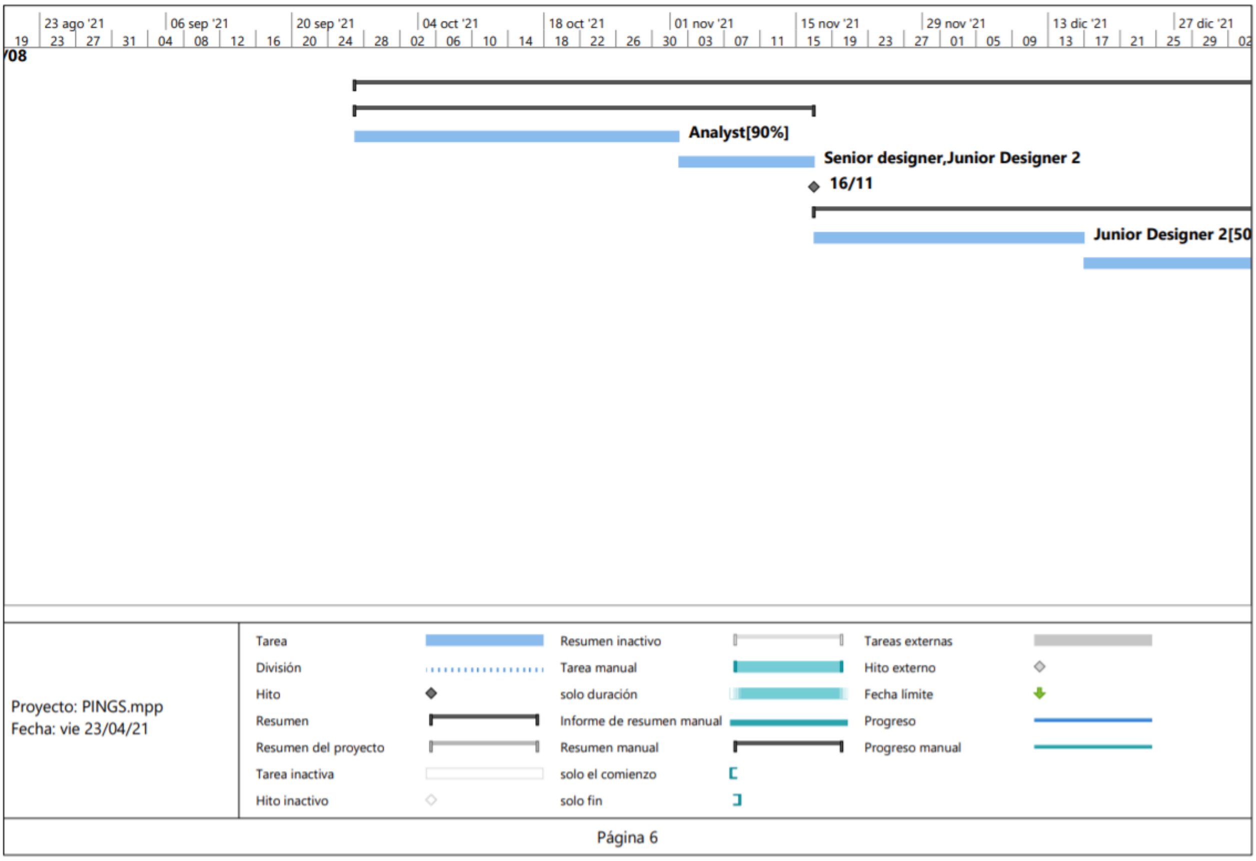


Figure 5. Gantt chart 2. Second part of the first increment.



Proyecto: PINGS.mpp
Fecha: vie 23/04/21

Página 6

Figure 6. Gantt chart 3. First part of the second increment.

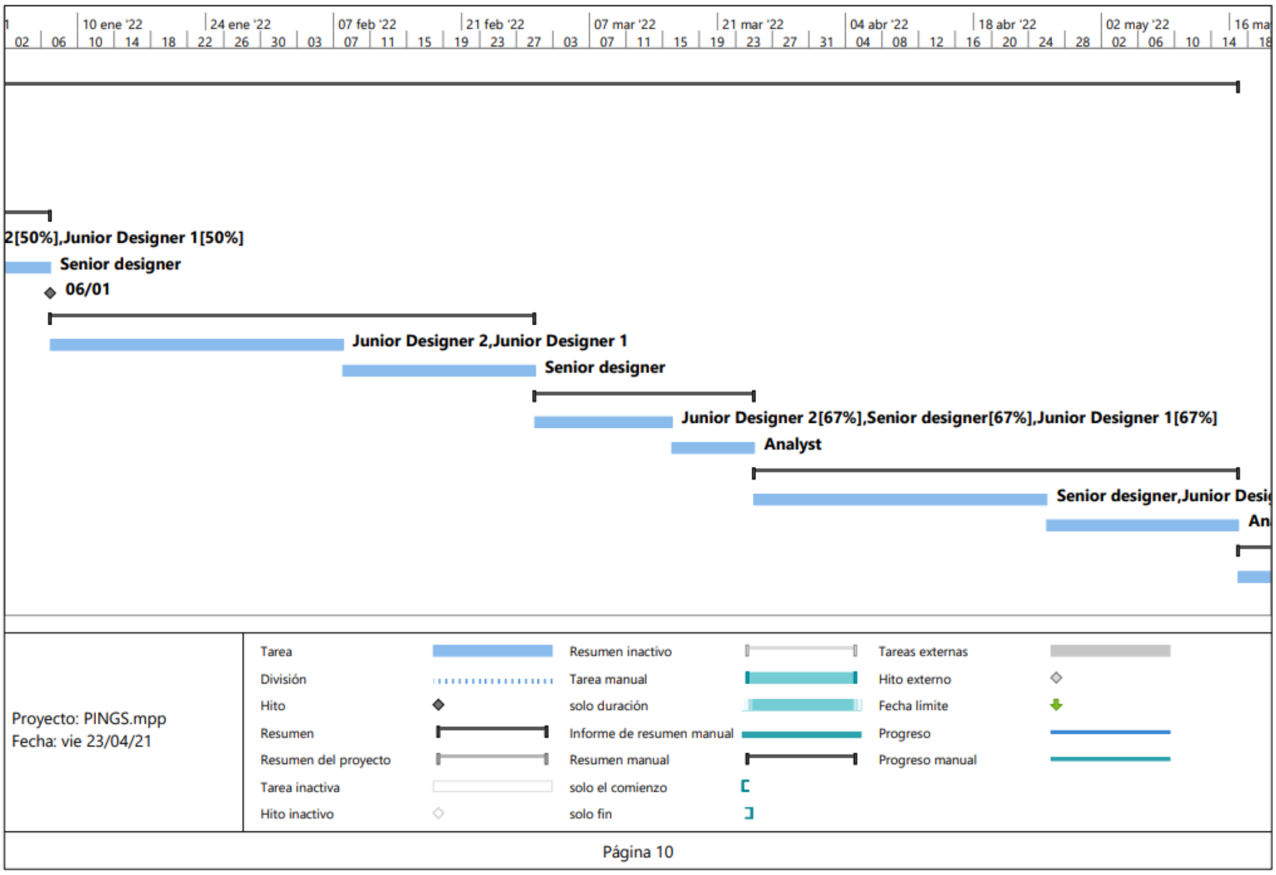
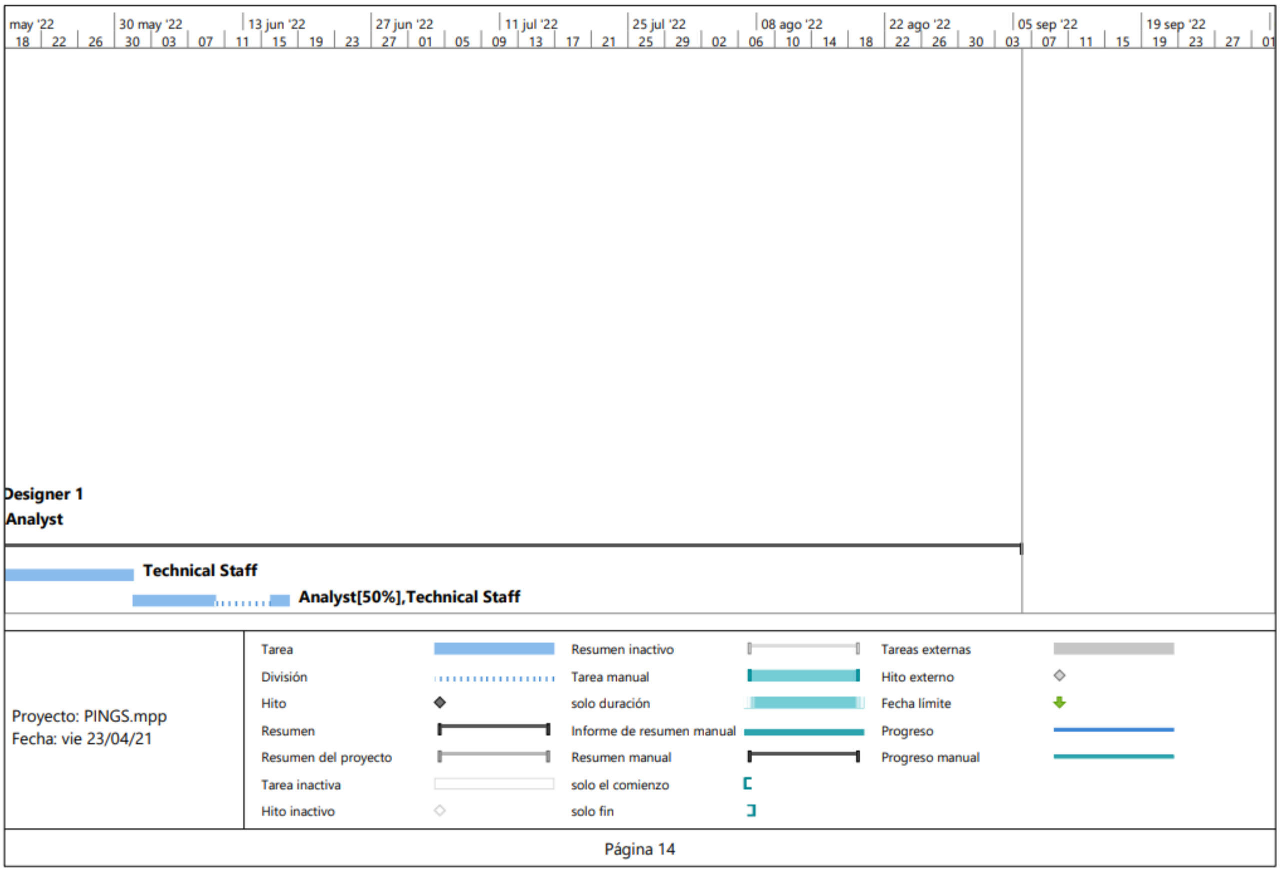


Figure 7. Gantt chart 4. Second part of the second increment.



Proyecto: PINGS.mpp
Fecha: vie 23/04/21

Página 14

Figure 8. Gantt chart 5. First part of the third increment.

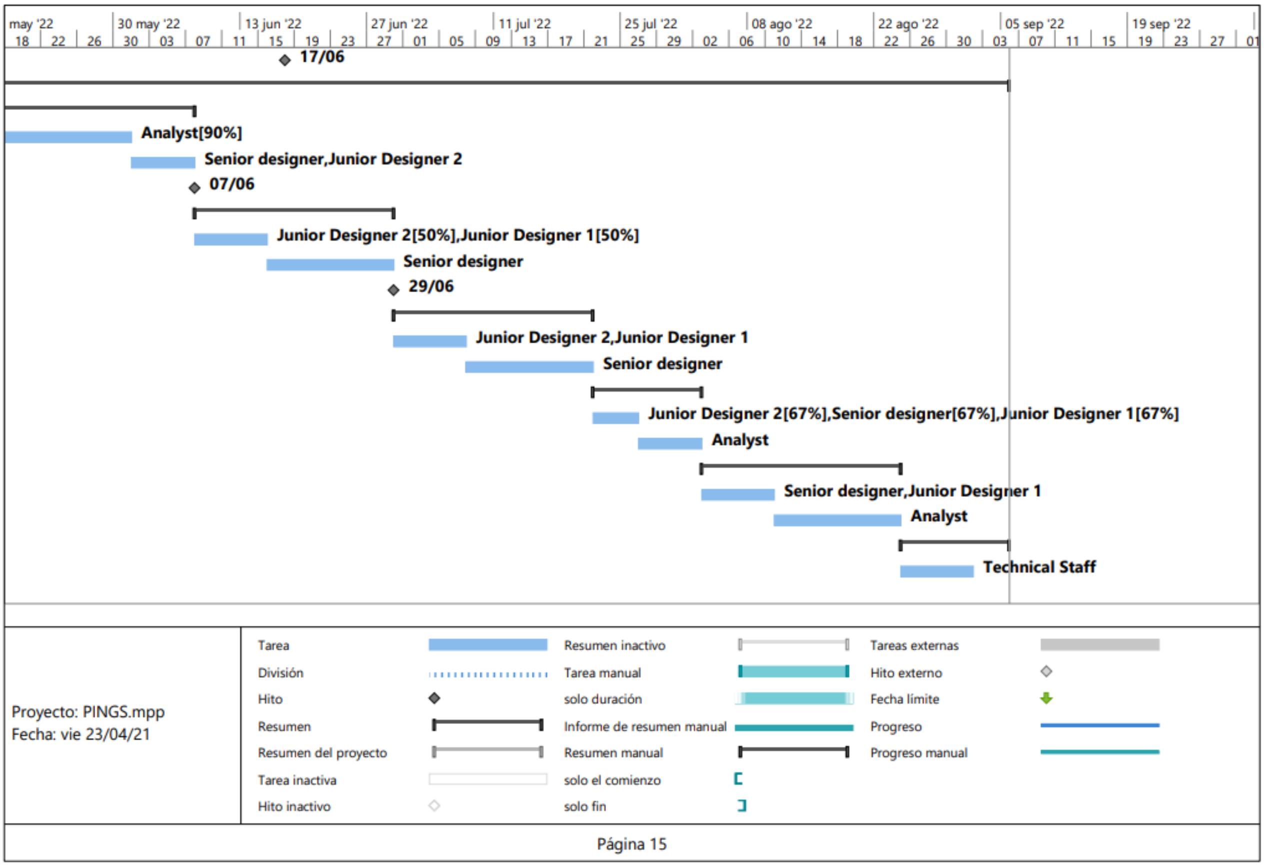


Figure 9. Gantt chart 6. Second part of the third increment.

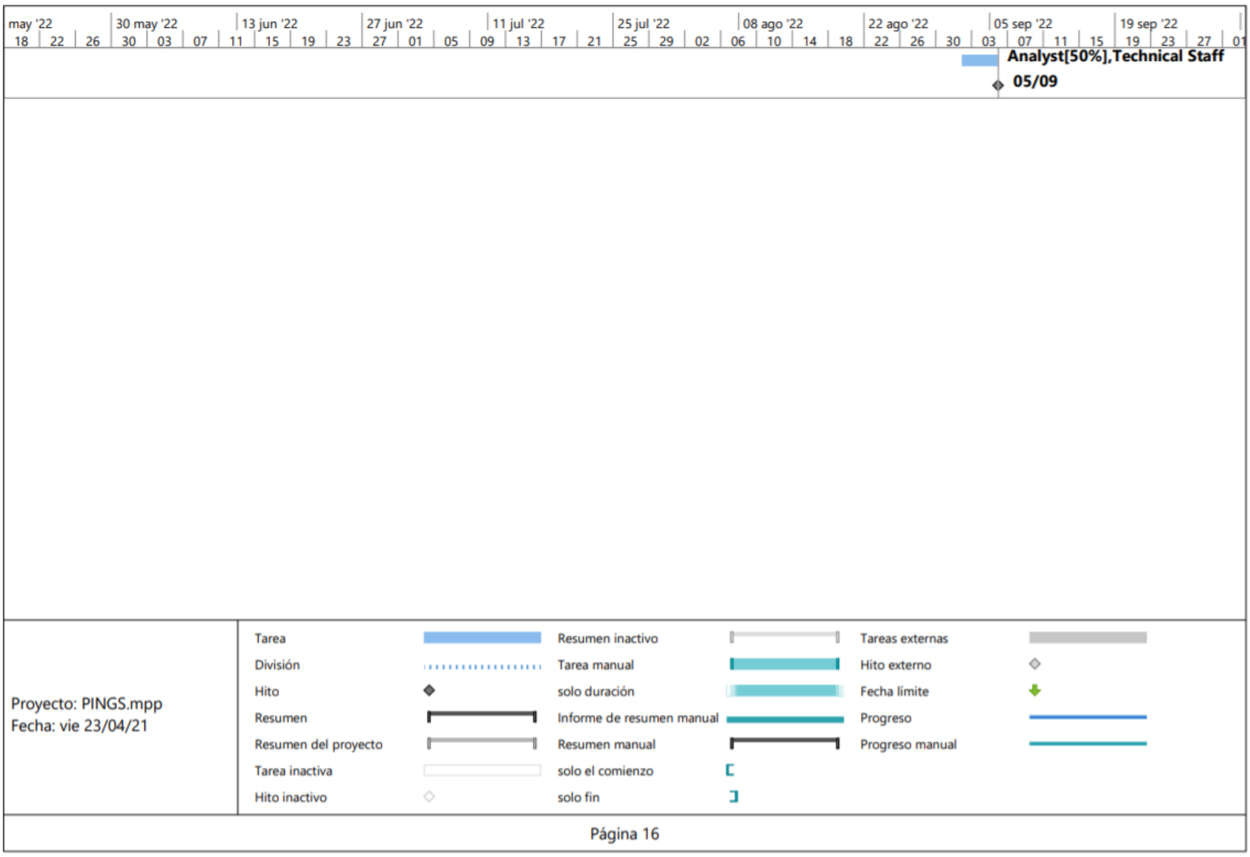


Figure 10. Gantt chart 7. Last part of the third increment

Annex D. Meeting Minutes

MEETING 1

MEETING ANNOUNCEMENT

From: S. Xiaofan
To: Carlos García, Miguel Arnaiz, Pablo Almarza, Samai García

DATE AND TIME: March the 24th 13:00
PLACE: MS Teams
DURATION: 1h
PURPOSE: Understand the purpose of the practice and organising the tasks.

AGENDA
Read the Assignment 2 and the Procedure Practical Assignment (15 min)
Talk about the conclusions and ideas (30 min)
Deciding who does each task (15 min)

DECISION FOLLOW-UP
We decided the team organisation and who acts as the different participants of the project plan.

MEETING MINUTES

DATE AND TIME: March the 24th 13:00
PARTICIPANTS: Pablo Almarza, Miguel Arnaiz, Carlos García, Samai García and S. Xiao Fernandez

KEY POINTS DISCUSSED
Structure of the Project Plan and scheduling the tasks

DECISIONS MADE

ACTIONS	RESPONSIBLE PERSON	DEADLINE
Start the project plan	Pablo Almarza Miguel Arnaiz Carlos García Samai García S. Xiao Fernández	Due 31/03/2021

MEETING 2

MEETING ANNOUNCEMENT

From: Samai García
To: Carlos García, Miguel Arnaiz, Pablo Almarza, S. Xiaofan

DATE AND TIME: March the 31st 13:00
PLACE: MS Teams
DURATION: 30 minutes
PURPOSE: Redefine the requirements catalogue and correct the errors from the previous practice. Decide which member of the team will do which subsystem function point based estimation, as there are 5 subsystems and we are 5 members.

AGENDA

Discussing about the new requirements and how to improve the previous ones (15 min)
Talk about the conclusions and ideas (15 min)

DECISION FOLLOW-UP

Carlos and Miguel will correct the requirements catalogue, while Xiao, Pablo and Samai will start with the Function Point method.

MEETING MINUTES

DATE AND TIME: March the 31st 13:00
PARTICIPANTS: Pablo Almarza, Miguel Arnaiz, Carlos García, Samai García and S. Xiao Fernandez

KEY POINTS DISCUSSED

Requirements catalogue, reorganization of Function Point method for the different subsystems.

DECISIONS MADE

ACTIONS	RESPONSIBLE PERSON	DEADLINE
Requirements catalogue	Miguel Arnaiz Carlos García Pablo Almarza	Due 07/04/2021
Function points	Samai García S. Xiao Fernández	Due 07/04/2021

MEETING 3

MEETING ANNOUNCEMENT

From: Pablo Almarza
To: Carlos García, Miguel Arnaiz, S. Xiaofan, Samai García

DATE AND TIME: April the 7th 13:00
PLACE: MS Teams
DURATION: 30 minutes
PURPOSE: Review the work done for the catalogue requirements and the function points, discuss about the problems that could have arise.

AGENDA

Review the adjustment factor of each subsystem and see if the results are realistics (15 min)
Resolve doubts and share ideas (15 min)

DECISION FOLLOW-UP

We will start redacting the Project Plan. Carlos and Miguel will start with Annex A, justifying the transactional function types for each subsystem, while Xiao and Samai will start with the estimation of the size of the system and Pablo will start creating the Gantt chart.

MEETING MINUTES

DATE AND TIME: April the 7th 13:00
PARTICIPANTS: Pablo Almarza, Miguel Arnaiz, Carlos García, Samai García and S. Xiao Fernandez

KEY POINTS DISCUSSED

Revision of the work done and start of the project plan.

DECISIONS MADE

ACTIONS	RESPONSIBLE PERSON	DEADLINE
Annex A	Miguel Arnaiz Carlos García	Due 21/04/2021
Estimation of the system size	Samai García S. Xiao Fernández	Due 21/04/2021

Gantt chart and graphs	Pablo Almarza	Due 21/04/2021

MEETING 4

MEETING ANNOUNCEMENT

From: Samai García
To: Carlos García, Miguel Arnaiz, Pablo Almarza, S. Xiaofan

DATE AND TIME: April the 21st 13:00
PLACE: MS Teams
DURATION: 1 hour
PURPOSE: Review project plan and fix possible errors.

AGENDA
Discussing the contents of the document (15 min)
Review the gantt chart and the result of the estimations (15 min)
Talk about the conclusions (15 min)
Divide the changes to make (15 min)

DECISION FOLLOW-UP
Carlos and Miguel will correct the errors in the requirements, Xiao, Pablo and Samai will correct possible errors in the project plan.

MEETING MINUTES

DATE AND TIME: April the 21st 13:00
PARTICIPANTS: Pablo Almarza, Miguel Arnaiz, Carlos García, Samai García and S. Xiao Fernandez

KEY POINTS DISCUSSED
Review and improve, if needed, the final document.

DECISIONS MADE

ACTIONS	RESPONSIBLE PERSON	DEADLINE
Requirements catalogue	Miguel Arnaiz Carlos García	Due 23/04/2021
Project Plan	Samai García S. Xiao Fernández Pablo Almarza	Due 23/04/2021