**PLANNING AND EXECUTION OF THE PROJECT BY USING UDP**

At the end of this document, we will be able to answer these three questions:

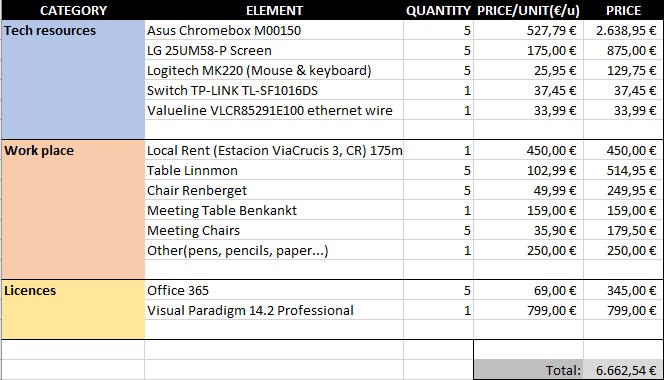
**1 How long will the project take?**

**2 When will the project will finish?**

**3 How much will the project cost?**

**FUNCTIONAL REQUIREMENTS**

Firstly, we have to be prepared to develop the project. We need some software and technology to start:



Then, we have noted down the functional requirements for the client and the server we are going to implement.

|  |  |
| --- | --- |
| CFR | Functional Requirement |
| CFR.1. | It should allow the management and the creation and login of users. |
| CFR.2. | It should support the acquisition of songs or albums, implying also manage payments. |
| CFR.3. | It should support playback of songs and albums, in streaming, from files stored on the device when this allows storage. |
| CFR.4. | It should support deletion and modification of lists of songs. |

|  |  |
| --- | --- |
| SFR | Functional Requirement |
| SFR.01. | It should support the addition of new songs and albums with their corresponding prices to be purchased individually as in the album in which have been published. |
| SFR.02. | It should support the modification of the metadata of the songs, as well as the price. |
| SFR.03. | It should support the deletion of songs and albums. |
| SFR.04. | It should support the search for songs by albums, artists, or title. |
| SFR.05. | It should support the automatic and manual creation of users. |
| SFR.06. | It should support the purchase of products (albums or songs) by users, keeping a list of who bought what, and allow payments through payment networks (PayP type \* l) |
| SFR.07. | It should support the Elimination of users. |
| SFR.08. | It should support sending messages to users. |

**USE CASE PRIORITY TABLE**

For simplicity, one functional requirement will be mapped into one use case:

**FR-----UC**

**1-----1**

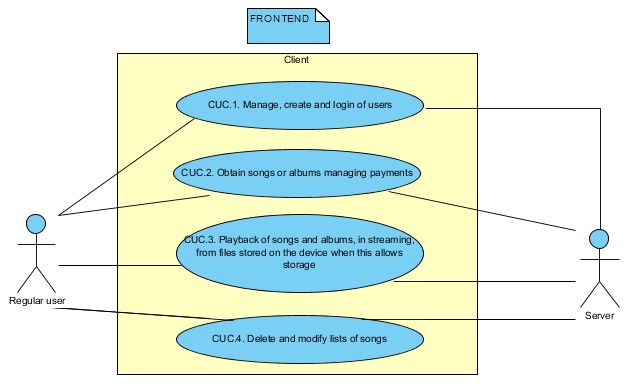
1-----n

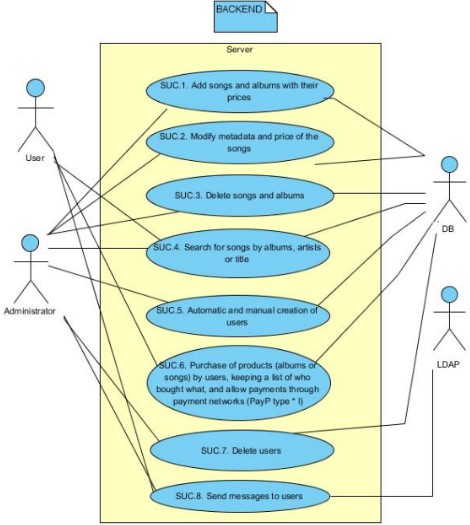
n-----1

n-----m

|  |  |  |
| --- | --- | --- |
| CFR | CUC | Use Case |
| CFR.1. | **CUC.1.** | Manage, create and login of users |
| CFR.2. | **CUC.2.** | Obtain songs or albums managing payments. |
| CFR.3. | **CUC.3.** | Playback of songs and albums, in streaming, from files stored on the device when this allows storage. |
| CFR.4. | **CUC.4.** | Delete and modify lists of songs. |

|  |  |  |
| --- | --- | --- |
| SFR | SUC | Use Case |
| SFR.01. | **SUC.1.** | Add songs and albums with their prices. |
| SFR.02. | **SUC.2.** | Modify metadata and price of the songs. |
| SFR.03. | **SUC.3.** | Delete songs and albums. |
| SFR.04. | **SUC.4.** | Search for songs by albums, artists or title. |
| SFR.05. | **SUC.5.** | Automatic and manual creation of users. |
| SFR.06. | **SUC.6.** | Purchase of products (albums or songs) by users, keeping a list of who bought what, and allow payments through payment networks (PayP type \* l) |
| SFR.07. | **SUC.7.** | Delete users. |
| SFR.08. | **SUC.8.** | Send messages to users. |



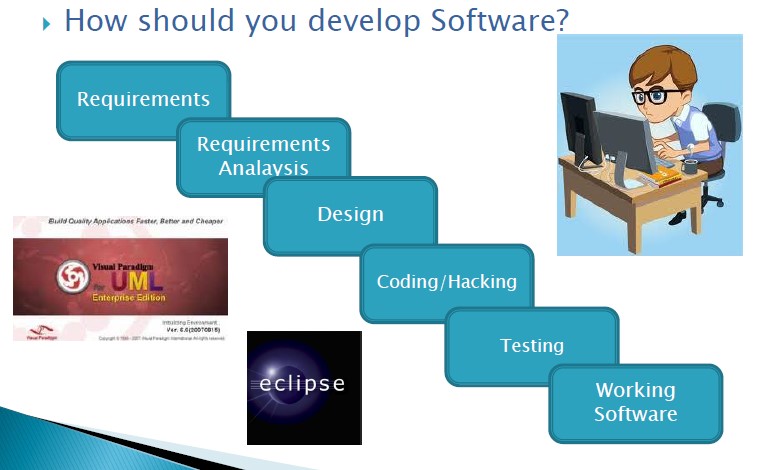


|  |  |
| --- | --- |
| P | SUC |
| 1 | SUC.1. |
| 2 | SUC.5. |
| 3 | SUC.2. |
| 4 | SUC.6. |
| 5 | SUC.3. |
| 6 | SUC.7. |
| 7 | SUC.4. |
| 8 | SUC.8. |

The use of cases are established in a priority order due to there are functionalities which can’t be generated until another one has been completed before because it depends on this.

|  |  |
| --- | --- |
| P | CUC |
| 1 | CUC.1. |
| 2 | CUC.2. |
| 3 | CUC.4. |
| 4 | CUC.3. |

We have decided this order in the priorities because we think that the most important parts are the songs and the users, so first we add them and later they can be modified or deleted.

We have estimated project’s duration in hours in every part of the process for every use of case. With the following estimations, we will be able to estimate the total time to realize the project and the cost.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| SUC | Priority | Requirements | Analysis | Design | Implementation | Testing |
| SUC.1. | 1 | 45 min | 60 min | 60 min | 90 min | 45 min |
| SUC.2. | 3 | 30 min | 45 min | 60 min | 75 min | 30 min |
| SUC.3. | 5 | 15 min | 15 min | 15 min | 30 min | 15 min |
| SUC.4. | 7 | 30 min | 45 min | 75 min | 75 min | 45 min |
| SUC.5. | 2 | 30 min | 30 min | 60 min | 45 min | 30 min |
| SUC.6. | 4 | 45 min | 60 min | 75 min | 105 min | 60 min |
| SUC.7. | 6 | 15 min | 30 min | 30 min | 30 min | 15 min |
| SUC.8. | 8 | 15 min | 15 min | 30 min | 45 min | 15 min |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| CUC | Priority | Requirements | Analysis | Design | Implementation | Testing |
| CUC.1. | 1 | 45 min | 45 min | 75 min | 90 min | 30 min |
| CUC.2. | 2 | 45 min | 60 min | 75 min | 90 min | 45 min |
| CUC.3. | 4 | 30 min | 60 min | 90 min | 90 min | 60 min |
| CUC.4. | 3 | 15 min | 15 min | 15 min | 30 min | 15 min |

**FUNCTIONAL GROUP**

We will join the client and the server with a similar function.

2 cycle ---> 12UC

**1 cycle ---> 8 UC**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| UC | SUC | CUC | P | R | A | D | I | T |
| UC1 | SUC.1. |  | 1 | 45 min | 60 min | 60 min | 90 min | 45 min |
| UC2 | SUC.5. | CUC.1. | 2 | 30 min + 45 min | 30 min + 45 min | 60 min + 75 min | 45 min + 90 min | 30 min + 30 min |
| UC3 | SUC.2. |  | 3 | 30 min | 45 min | 60 min | 75 min | 30 min |
| UC4 | SUC.6. | CUC.2. | 4 | 45 min + 45 min | 60 min + 60 min | 75 min + 75 min | 105 min + 90 min | 60 min + 45 min |
| UC5 | SUC.3. | CUC.4. | 5 | 15 min + 15 min | 15 min + 15 min | 15 min + 15 min | 30 min + 30 min | 15 min + 15 min |
| UC6 | SUC.7. |  | 6 | 15 min | 30 min | 30 min | 30 min | 15 min |
| UC7 | SUC.4. | CUC.3. | 7 | 30 min + 30 min | 45 min + 60 min | 75 min + 90 min | 75 min + 90 min | 45 min + 60 min |
| UC8 | SUC.8. |  | 8 | 15 min | 15 min | 30 min | 45 min | 15 min |

**UDP PHASES**

**Inception Phase:**

iteration 0: Determine life cycle goals

Duration: 9 h

**Elaboration Phase:** We will obtain the system architecture

iteration 1: UC1

Duration: 5 h

iteration 2: UC2

Duration: 8 h

iteration 3: UC3

Duration: 4 h

**Construction Phase:** Initial working capability

iteration 4: UC4

Duration: 11 h

iteration 5: UC5

Duration: 3 h

iteration 6: UC6

Duration: 2 h

iteration 7: UC7

Duration: 10 h

iteration 8: UC8

Duration: 2 h

**Transition Phase:**

iteration 9: We establish the last reviews and conditions.

Duration: 18 h

|  |  |
| --- | --- |
| Iteration | Use Cases |
| it1 | Add songs and albums with their prices. |
| it2 | Automatic and manual creation of users.  Manage, create and login of users |
| it3 | Modify metadata and price of the songs. |
| it4 | Purchase of products (albums or songs) by users, keeping a list of who bought what, and allow payments through payment networks (PayP type \* l)  Obtain songs or albums managing payments. |
| it5 | Delete songs and albums.  Delete and modify lists of songs. |
| it6 | Delete users. |
| it7 | Search for songs by albums, artists or title.  Playback of songs and albums, in streaming, from files stored on the device when this allows storage. |
| it8 | Send messages to users. |

**RESULTS IN EACH PHASE**

**Inception Phase**

This phase consists in the development of a description of the product since a good specification of the requisites.

* Identify and prioritize the most important risks.
* Plan the next phases.
* Estimate project’s duration in an approximate way.

**Elaboration Phase**

We will elaborate the most critical and important use cases that we identify in the previous phase and as a result we obtain an architectural base line. At the end of this phase we will be able to plan the activities and estimate the resources we will need to finish the project.

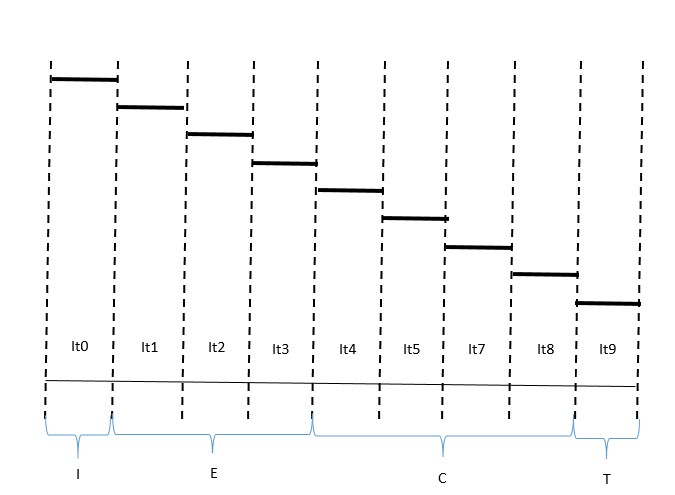
**Construction Phase**

It consists in the elaboration of a product which contains all use cases that we have agreed in this version. We start from the architectural base line to the complete system using the resources. If we find some error, in the next phase they will be resolve.

**Transition Phase**

The product will be the beta version. We will correct the problems and errors and some improvements to the final version.

**ITERATIONS**



Each component of the group works in a team together and organized. We have the same responsibilities in each iteration because all together are in all the parts of the software development.

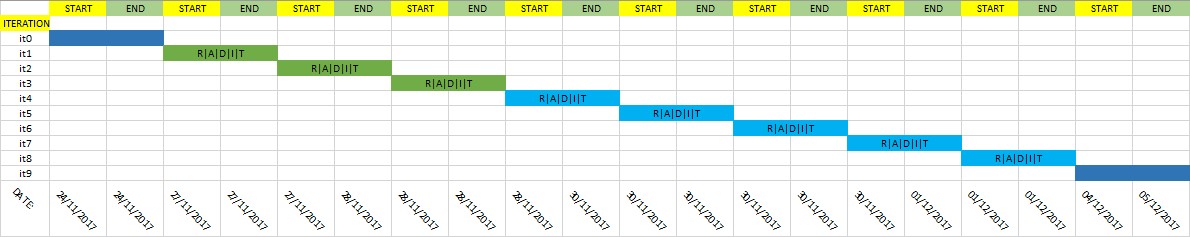
All the components of the project have a salary of 6€/hour. The components are:

* Adrián Ollero Jimenez
* Beka Bekeri
* Daniele Acquaviva
* Enrique Garrido Pozo
* Pablo Mora Herreros

In the iteration 0 we have and estimated cost of 1000€ whereas in the last iteration the estimated cost is 2000€.

To calculate the rest of the cost, we have to estimate the duration of the project in order to calculate the hours all components we are working.

We know that from Monday to Friday the working day are 9 hours.

**GRANT GRAPH**

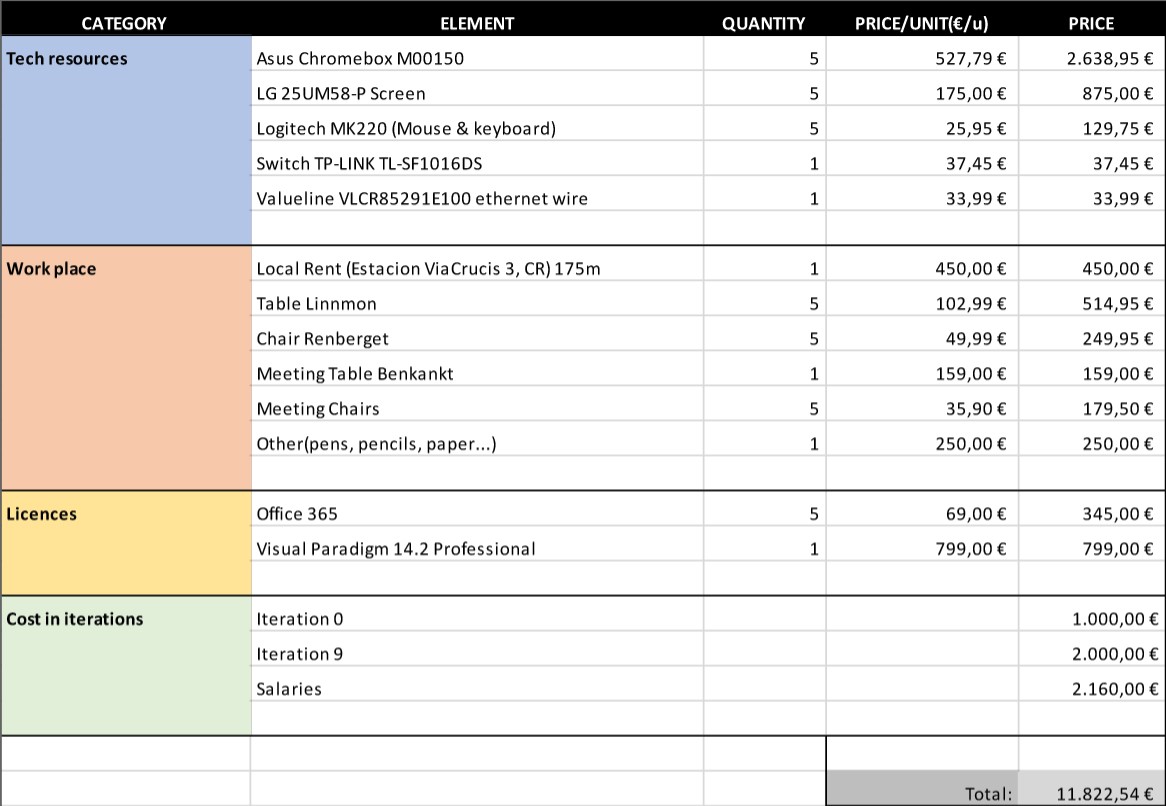
**TOTAL COST**

At the end of the project, each component will have worked 8 days, so:

8x9=72 hours in total for each component

72x5x6=2160€ for salaries

TOTAL COST: 6662,54+1000+2160+2000=11822,54€



**CONCLUSION**

After estimating all the calculations and considerations, we can say:

**1** The project will go on for **8 working days**, from **November 24th to December 5th.**

**2** The project will finish on **December 5th**.

**3** The project will cost **11822,54€.**