BOSST-GARREN TALDEA

**Authors: Abde, Jon and Beñat**

**Date: 5/3/2021**

INDEX

[1. PROJECT CONTEXT 4](#_Toc72319502)

[2. APPLICATION ANALISYS 4](#_Toc72319503)

[a) WEB 4](#_Toc72319504)

[b) Java application 5](#_Toc72319505)

[3. APPLICATION DESIGN 5](#_Toc72319506)

[A) Use case diagrams 5](#_Toc72319507)

[B) Activity diagrams 7](#_Toc72319508)

[WEB 7](#_Toc72319509)

[JAVA 8](#_Toc72319510)

[C) Class diagram 8](#_Toc72319511)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Duration (hours)** | **Beginning date** | **Finishing date** | **Responsible** |
| 1.Kontratua | 2 | 03-05-2021 | 03-05-2021 | Jon |
| 2. 3\_eranskina 1-4 | 37 | 03-05-2021 | 07-05-2021 | Beñat, Abde, Jon |
| 2.1 Project context | 1 | 03-05-2021 | 03-05-2021 | Beñat |
| 2.2 Project analisys | 6 | 03-05-2021 | 05-05-2021 | Beñat, Abde, Jon |
| 2.3 Application design | 30 | 05-05-2021 | 07-05-2021 | Beñat, Abde, Jon |
| 2.3.1 USE-CASE Diagram | 2 | 07-05-2021 | 07-05-2021 | Beñat, Abde, Jon |
| Mikel | 2 | 05-05-2021 | 05-05-2021 |  |
| Abde | 2 | 05-05-2021 | 05-05-2021 |  |
| Benat | 2 | 05-05-2021 | 05-05-2021 |  |
| Jon | 3 | 05-05-2021 | 05-05-2021 |  |
| 2.3.2 Class Diagram | 2 | 05-05-2021 | 07-05-2021 | Beñat, Abde, Jon |
| 2.3.3 Activity Diagram | 2 | 07-05-2021 | 07-05-2021 | Beñat, Abde, Jon |
| Mikel | 2 | 05-05-2021 | 05-05-2021 |  |
| Abde | 3 | 05-05-2021 | 05-05-2021 |  |
| Benat | 3 | 05-05-2021 | 05-05-2021 |  |
| Jon | 3 | 05-05-2021 | 05-05-2021 |  |
| 2.3.4 E-R Diagram | 3 | 04-05-2021 | 05-05-2021 | Beñat, Abde, Jon |
| 2.4 TASK PLANNING |  | 05-05-2021 |  |  |
| 2.5 IMPLEMENTATION AND  INSTALLATION |  |  |  |  |
| 2.6 USERS GUIDE |  |  |  |  |
| 2.7 THOUGHTS |  |  |  |  |
| 2.8 BIBLIOGRAPHY |  |  |  |  |
| 3.START USING TRELLO | 1 | 04-05-2021 | 04-05-2021 | Beñat, Abde, Jon |
| 4.TECHNICAL POURPOSE | 1 | 06-05-2021 | 06-05-2021 | Beñat |
| 5.DB DESIGN | 3 | 04-05-2021 | 06-05-2021 | Beñat, Abde, Jon |
| 6.WEB MAP | 2 | 06-05-2021 | 06-05-2021 | Jon |
| 7.START WORKING WITH GIT | 2 | 05-05-2021 | 05-05-2021 | Beñat,Abde,Jon |
| 8.HAVING UPDATED THE DOCUMENTATION |  | 10-05-2021 | 14-05-2021 | Jon |
| 9.DOCUMENT CODE | 5 | 11-05-2021 | 14-05-2021 | Abde,Jon |
| 10.JAVA ALPHA VERSION | 28 | 10-05-2021 | 14-05-2021 | Abde,Beñat |
| 11.TECHNICAL TESTS |  |  |  |  |
| 12.DB SCRIPT | 4 | 07-05-2021 | 10-05-2021 | Beñat,Jon |
| 13.WEB ALPHA VERSION | 34 | 07-05-2021 | 14-05-2021 | Beñat,Jon |
|  |  |  |  |  |

**SOFTWARE APPLICATION IMPLEMENTATION/INSTALLATION**

**USERS GUIDE**

**THOUGHTS**

**BIBLIOGRAPHY**

1. PROJECT CONTEXT

We have to create an application for Durango beekeepers association named “Erlete” to manage the extractions in the association.

1. APPLICATION ANALISYS

This application will consist of two parts. On the one hand, it will be the web application and on the other hand, it will be the java application.

# WEB

In the web, there are going to be different options in the header bar which depending on the user, there are going to be some options or not.

* **For GUEST:** General info about association or association history: A little introduction about the association and all the users are allowed to see and read this part.
* **For PARTNERS:** This part is only for those who are members or who want to be members. In case that someone wants to register, he must fill a form and pay a membership of 30€ per year.
  1. Room booking: This option is only available for members. They will be able to book the extraction room to take out their honey. Once that they go to extract their honey, they could use some cans that are in the society for all the members, in case that the cans were available. Moreover, the room and the cans availability will be shown in order to know if they can be used or not. It’s strongly recommended to check cans availability the day before going to extract the honey due to the fact that some cans may not be available. In this case beekeepers should bring their own containers.
  2. Your bookings: This option, as the previous one, will be shown to partners. Here they will be able to see the bookings they have made. As a result, they can manage their bookings.
     + If the booking is previous than the current day, it will show a button to complete a form. Here they must select the cans they have used and que quantity extracted. Once they complete it, it will be completed.
     + If the day is not actual or previous, they would be able to cancel their booking.

Once they have selected the cans, this will be unavailable for 14 days.

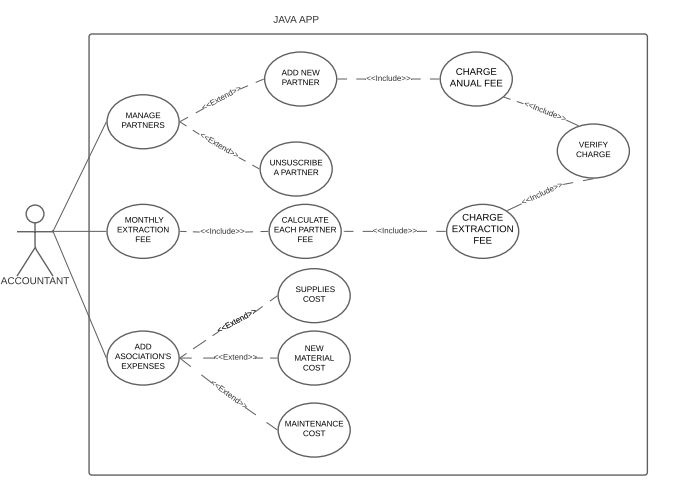
# Java application

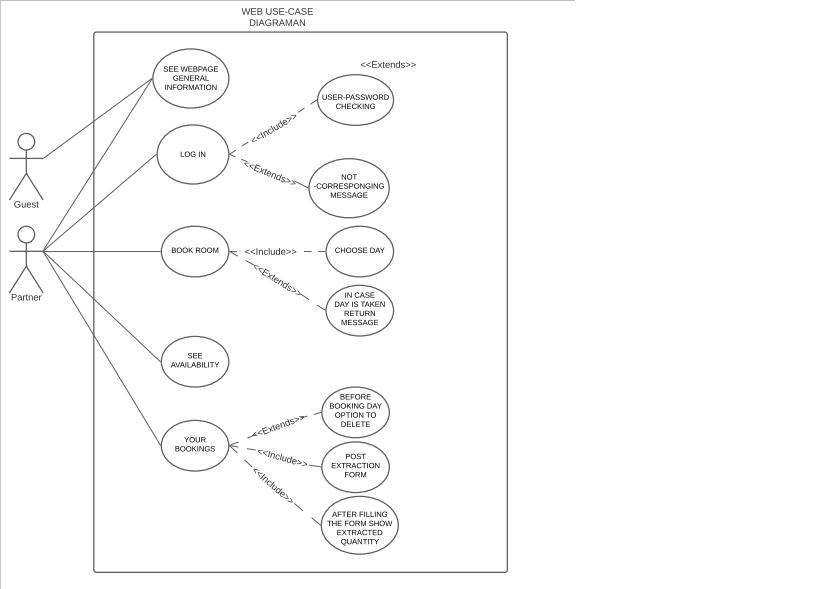
In the java application, the only one who is going to use it is the accountant. This are all the options available to use in the application:

* 1. He will be able to manage the annual fees of the members. In case that someone don not want to keep as a member, the accountant must to deny the annual payment.
  2. Depending on how much honey do the members take, the application will calculate the total money to pay. (We thought here we could create a Frame, which has a form, and in that form, there would be a button that it will calculate the total price depending on the user. Once the button is pushed, a TRIGGER will activate and the user name, the month when the members tokes the honey and the total price will be saved in a database table.)
  3. The materials, the maintenances and the supplies, which are paid during the year, will be managed. Moreover, there is a table to save all the information in the database. As a result, there are two options:
     + The first one is going to allow the accountant to see how much they have paid.
     + The second option, the accountant can insert new expenses into the database table.

1. APPLICATION DESIGN

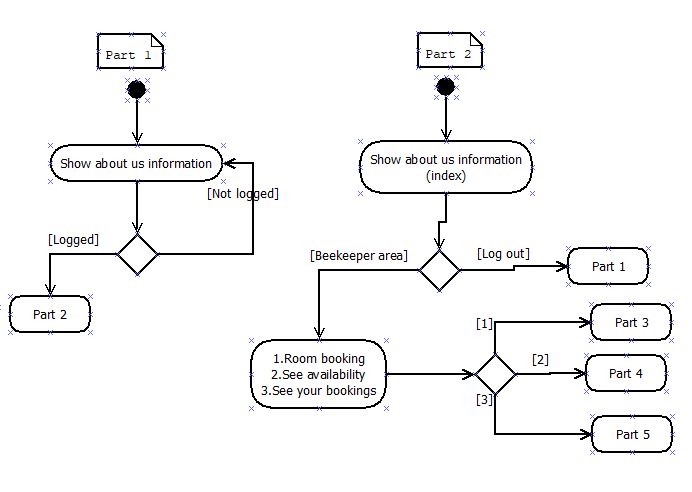
A) Use case diagrams

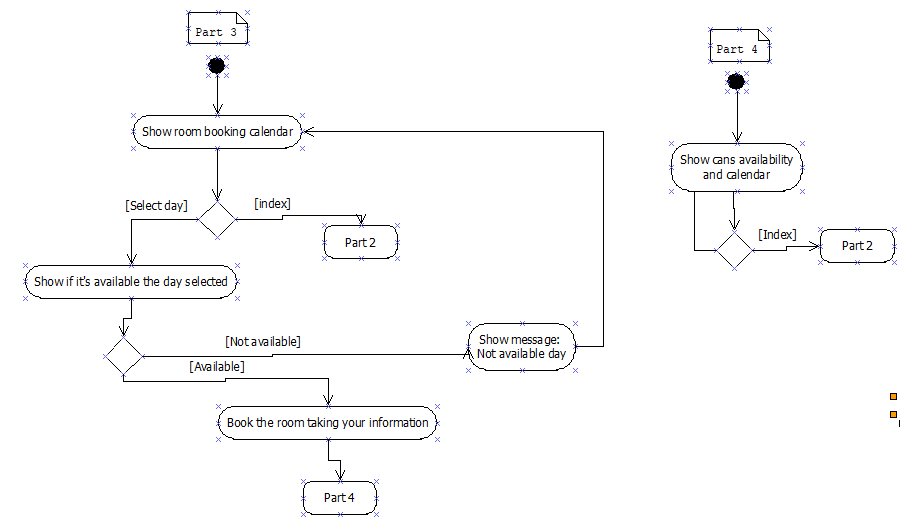


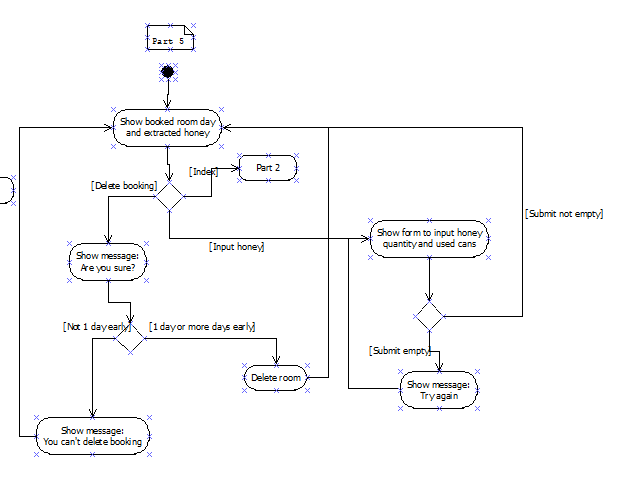


B) Activity diagrams

WEB







JAVA

C) Class diagram

