**INDEX**

1. **PROJECT CONTEXT AND ANALYSIS**
   1. **CONTEXT AND GENERAL ANALYSIS**
   2. **PROJECT IN RELATION WITH EACH SUBJECT**
      1. **PROGRAMMING**
      2. **DB**
      3. **MARKUP LANGUAGE**
      4. **DEVELOPMENT ENVIRONMENTS**
      5. **SYSTEMS**
2. **APPLICATION DESIGN**
   1. **TRELLO PLANIFication**
   2. **USE CASE DIAGRAM**
   3. **CLASS DIAGRAM**
   4. **DATABASE DESIGN**

**1.1 Contex and general analysis**

This challenge is designed around the creation of a management system for the uni Eibar-Ermua canteen. In this canteen, students can make reservations for a meal, and workers of said canteen should be able to see beforehand how many people reserved the canteen services at any given date.

Our job as developers is to create tools, in the form of a website and different applications to manage this canteen.

For the students' part, we have 2 types of students: Normal students and residents. These 2 types of users are going to be able to access the canteen system through the website and make reservations and several other actions on this website. These actions include but are not limited to: Cancel reservations, check their account balance, change their user data or notify allergies that they might have.

The workers on the other side have different actions that they should be able to perform, like: Checking number of reservations for a given date, checking the allergen information of the reserves, changing user data, creating new users and giving them privileges and updating the user balance. Some of these actions will be performed via the website while others will be done through either a CLI Python application or a GUI Java application.

For this whole project to function, we will also need to design and configure several services that will serve our programs to the user.

The challenge is related in different ways to each subject of our class, and we will have to make different things for each one of the subjects. Next we will analyze what are we going to have to develop for each subject:

**1.2.1 Programming**

In the case of the Java Programming subject, we will need to create a GUI application so that the canteen workers can receive payments from the users. This application will also be able to show user information and reservation information, as well as some statistics about the usage of the canteen.

We will use JavaFX to be able to accomplish this. The program will be connected to the database of the challenge and will perform actions to it like inserts to create a new transaction or delete to delete reservations, among others.

This application will only be used by canteen workers and therefore will not be accessible to normal users.

**1.2.2 Databases**

In the database subject we are tasked with designing a database that will be able to hold all the information that our canteen needs to function. The design of this database is very important, since it is the base of our project and all applications will be performing actions to this database.

Apart from the design of the database itself, we will need to create systems within the database to perform actions independently from the applications. We can achieve this by using triggers and procedures.

**1.2.3 Markup language**

In the subject of markup language, we will be creating a website so that users can connect to the canteen management system. This website will be available publicly to every kind of canteen user and will do different things depending on the type of user that is connected to it.

For example, a student that is a resident should be able to make periodic reservations, while their non-resident counterpart won’t be. The canteen workers will have the capability to see the number of diners for each date and to see if any of the users that have reserved the service are allergic to the menu. As an added bonus, we are thinking of creating a system in which workers will be able to create new menus with draggable menu items.

**1.2.4 Development environments**

In the DevEnv subject, we will have to create a CRUD, CLI application to manage users of the system. This application will perform actions in files, with structured data. This application will be directed towards workers of the canteen, so normal users will not be able to access it.

**1.2.5 Systems**

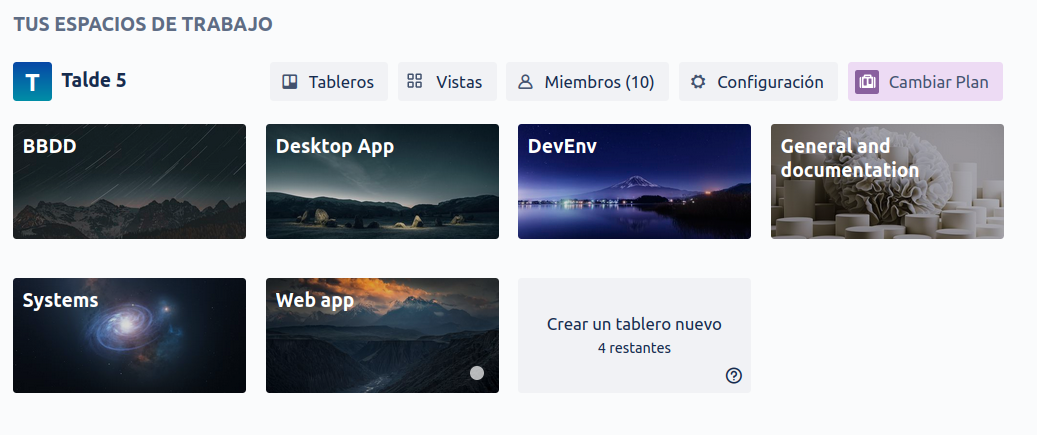
In this subject, we will create the servers and their different configurations so that our applications can function. After looking at what our application must be able to do, our server will need to include a database server, to hold our database, a web server with php and https enabled so that our website can work correctly, a DNS server to perform the IP to domain name translation.

This server or servers will hold all the services that the canteen management system will need.

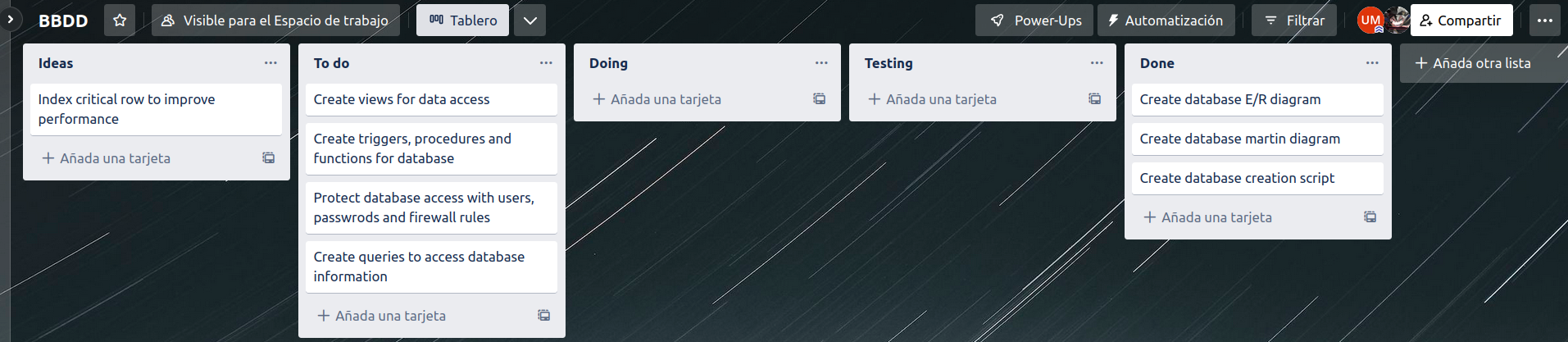
**2. Application design**

**2.1 Trello planification**

For the planification of our project we are using trello. Trello uses the kanban system of cards to hold the tasks that the project needs to continue. These cards can be assigned to different developers and can have deadlines to control the pace of development. We have created Trello accounts for our developers and created a workspace for the project. Inside this workspace we have created a board for each subject. This boards will hold the cards with the tasks necessary to complete the challenge.



The workspace is named “Talde 5”, which holds the boards.



An example of a board, in this case the database board.

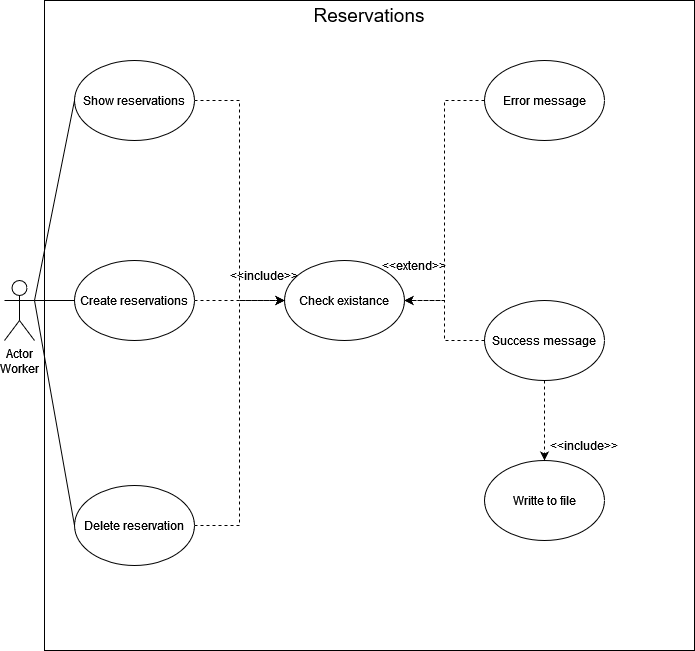
**2.2 Use case diagrams**

We will create use case diagrams to illustrate the flow of actions that a certain type of user can perform on a system. These diagrams are useful to visualize the actions that users can perform on our systems.

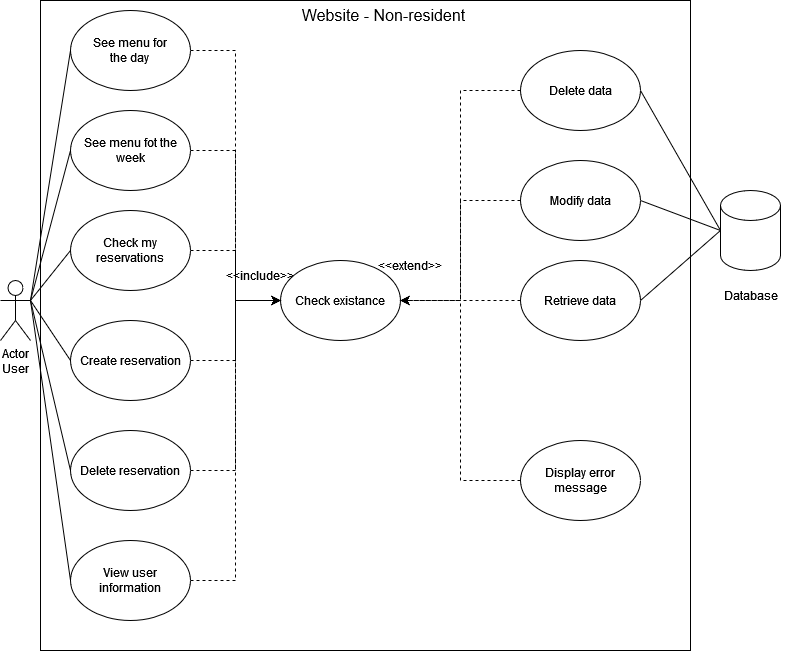
Use case diagrams:

- Python apps transition

* Python app - reservations



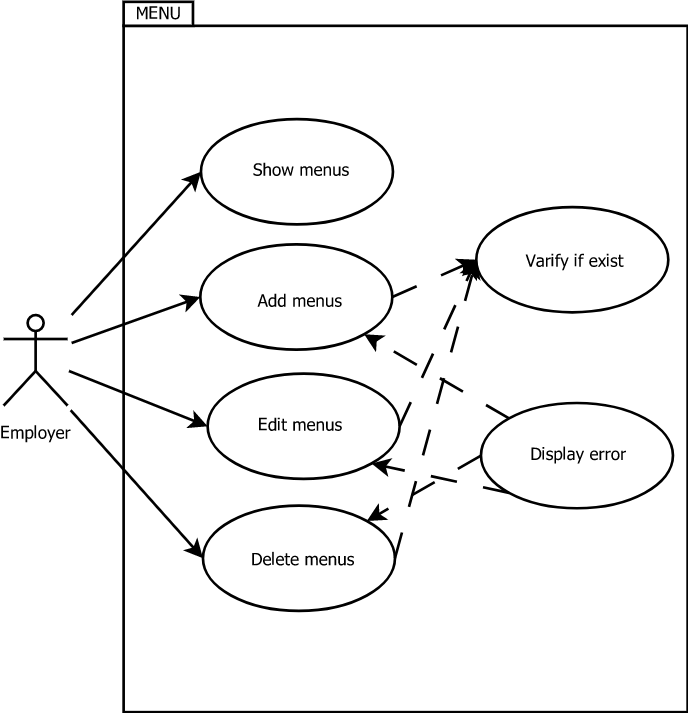
* Web app - Non resident



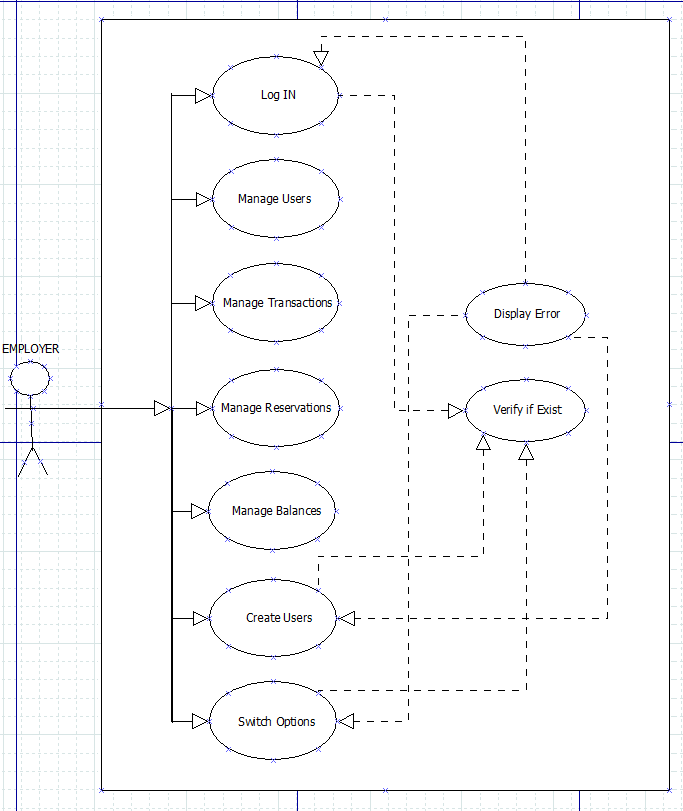
* Web app - Resident



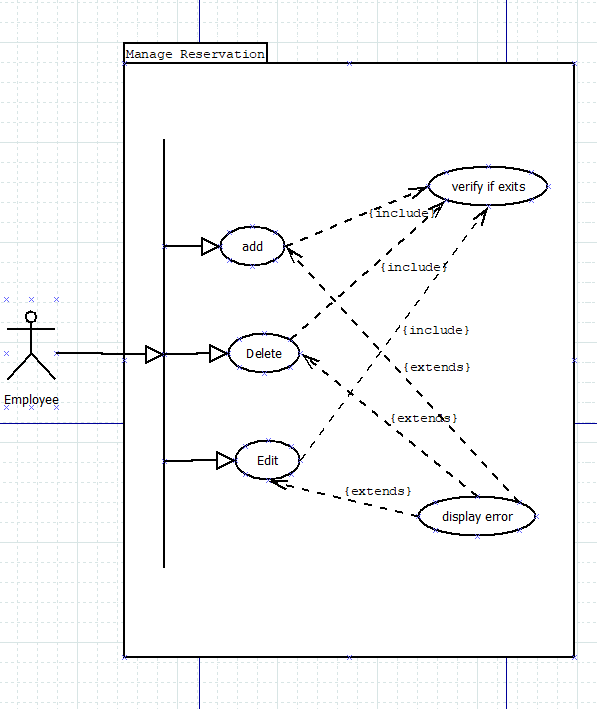
Menu:



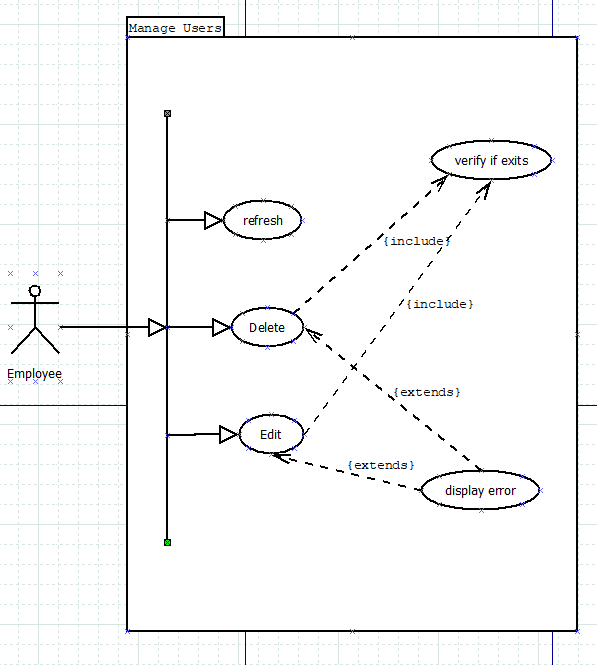
Desktop Aplication:



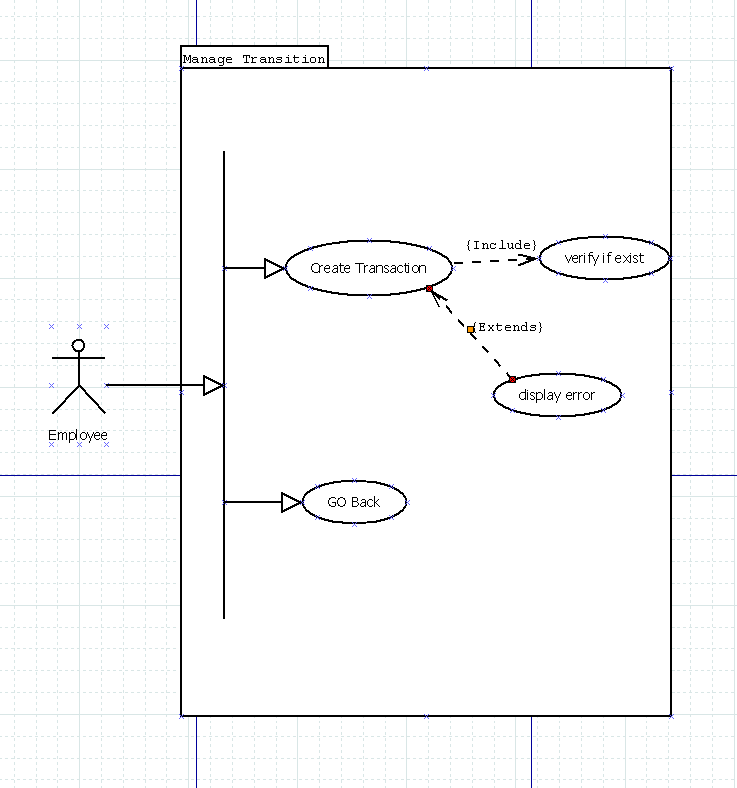
Manage Reservation:



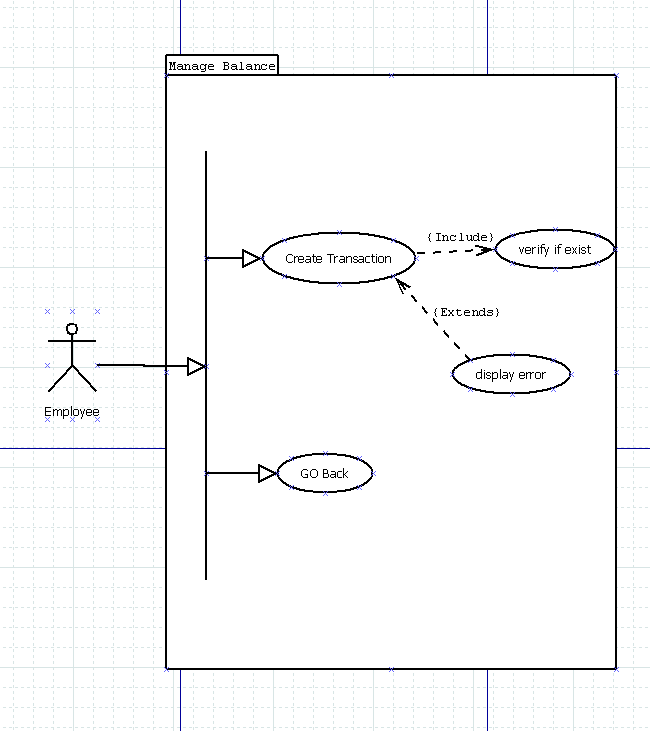
Manage Users:



Manage Transitions:



Manage Balance:



**2.3 Class diagram**

The class diagrams are created to show the relationship between objects in an OOP application. Since our Java and Python applications are going to be object oriented, we will have class diagrams for each one of these applications.

**2.4 Database design**

The database design is one of the most important parts of the challenge since our applications will connect to it and perform actions on it. That is why it’s design is very important for the challenge. For the design of the database, we started writing in a piece of paper and reading over the challenge description in order to understand which pieces of data the challenge would need.

We created different diagrams, each one as they thought the database would need to be, and then decided to select one of them.

After that we created a martin diagram of the selected database. This is the result:

