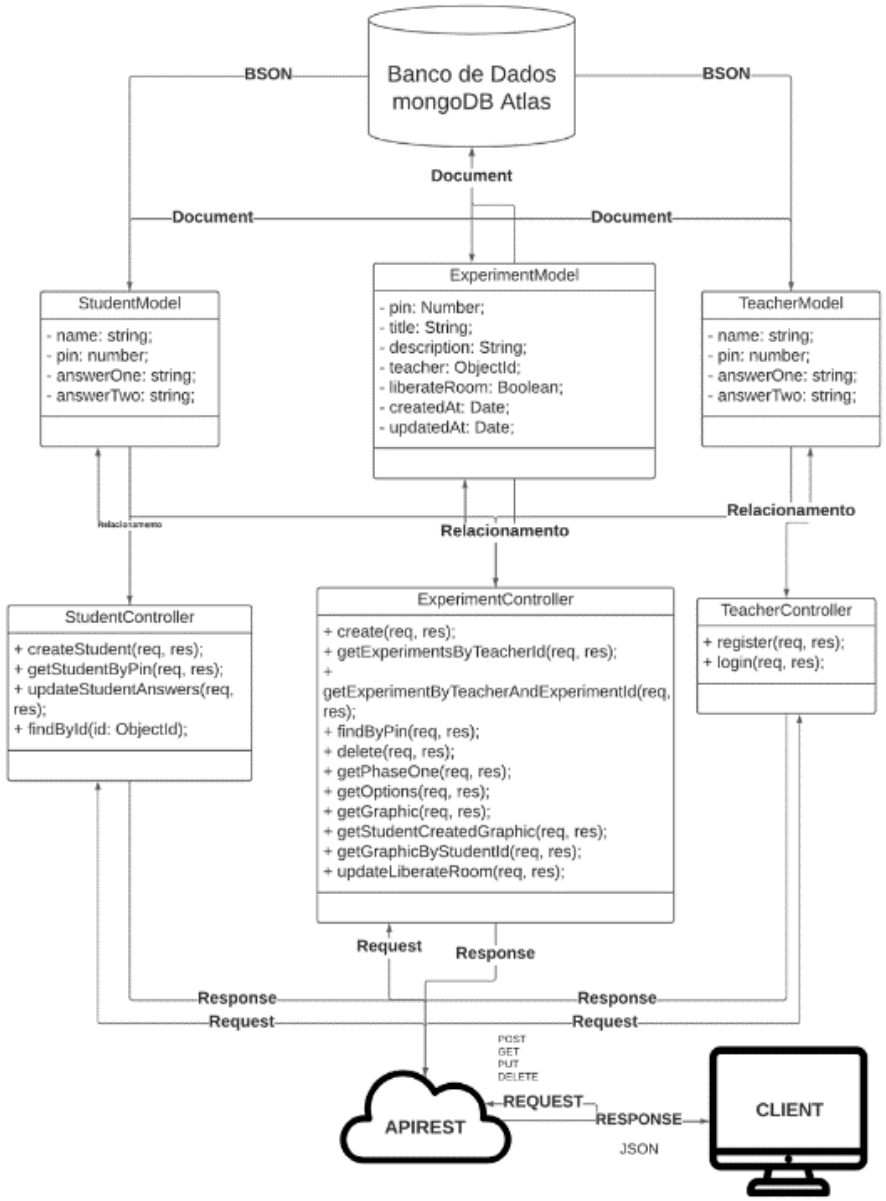
Documentation: Technical part of the project and step by step instructions on how to use all the features

Fernanda Bonfim Santos - fernandacomputerscience@gmail.com

Folder Structure & MVC

The project's folder architecture was organized according to the MVC (Model-View-Controller) pattern, ensuring a clear and modular structure. The folders were divided into models, controllers, routes, services, contexts, components, pages and utilities, each with its specific responsibility. This facilitated the maintenance, scalability and extensibility of the code throughout the project's development. Figure 1 presents a diagram that outlines how this architecture was designed, (Gama, N., & Campos, M. L. M.).

Figure 1 - MVC - Rest Api Diagram.



The backend structure was divided into three main components:

- Models: This folder contains the schemas and attributes of the objects used in the application. Models define the structure of data and its relationships in the database.
- Controllers: This is where the logical part responsible for manipulating data models resides. Two subfolders were created in this directory:
- Auth: Contains the controllers responsible for authenticating system users, including functions such as registration, login and data recovery.
- Experiment: Encompasses controllers related to experiments, such as creation, data manipulation and graph generation.
- Database: This directory handles the connection to the database. The conn.js file is responsible for connection attempts to MongoDB from a string stored in the .env file.

Project architecture

- The project's folder architecture was organized according to the MVC (Model-View-Controller) pattern, ensuring a clear and modular structure. The folders were divided into models, controllers, routes, services, contexts, components, pages and utilities, each with its specific responsibility. This facilitated the maintenance, scalability and extensibility of the code throughout the development of the project, (Richardson, L., & Amundsen, M.).
- The backend structure was divided into three main components:
- Models: This folder contains the schemas and attributes of the objects used in the application. Models define the structure of data and its relationships in the database.
- Controllers: This is where the logical part responsible for manipulating data models resides. Two subfolders were created in this directory:
- Auth: Contains the controllers responsible for authenticating system users, including functions such as registration, login and data recovery.
- Experiment: Encompasses controllers related to experiments, such as creation, data manipulation and graph generation.
- Database: This directory handles the connection to the database. The conn.js file is

responsible for connection attempts with MongoDB from a string stored in the .env file.

Organization of the Front-end Structure

The front-end folder structure was meticulously organized to optimize the development and maintenance of the project. The src directory represents the root file that houses all the raw content of the front-end project, while the files outside it are the default ones, including index.js.

1. Reusable Components

The components folder contains all the reusable components of the code. For example, BaseAuth is a component that contains the base style of the user authentication page, such as the background and card settings, as well as an area for inserting content. Each component is developed with an index.jsx file, which contains the React JavaScript code, and its own style file, styles.css.

2. Contexts and Providers

The context folder houses all the contexts and providers used in the application. Among them, the Users context stands out, which includes information relevant to the teacher, facilitating its reuse throughout the platform. In addition, there is the Authorization context, which manages the Boolean state of the button responsible for releasing the experiment results to the students.

3. Navigation Routes

The navigation routes are organized in the routes folder, created from the pages intended for user navigation. For better organization, two new subfolders were created: StudentPages and TeacherPages, which contain the JSX and CSS files for each page, clearly separating those accessed by the regular user and the teacher.

4. User Pages

The pages folder includes all the files created to build the user navigation pages. In order to properly separate the files for the pages intended for regular users and teachers, the StudentPages and TeacherPages folders were created. Within them, there are the JSX and CSS files for each page individually, ensuring a clear and intuitive organization of the structure.

Additional Details

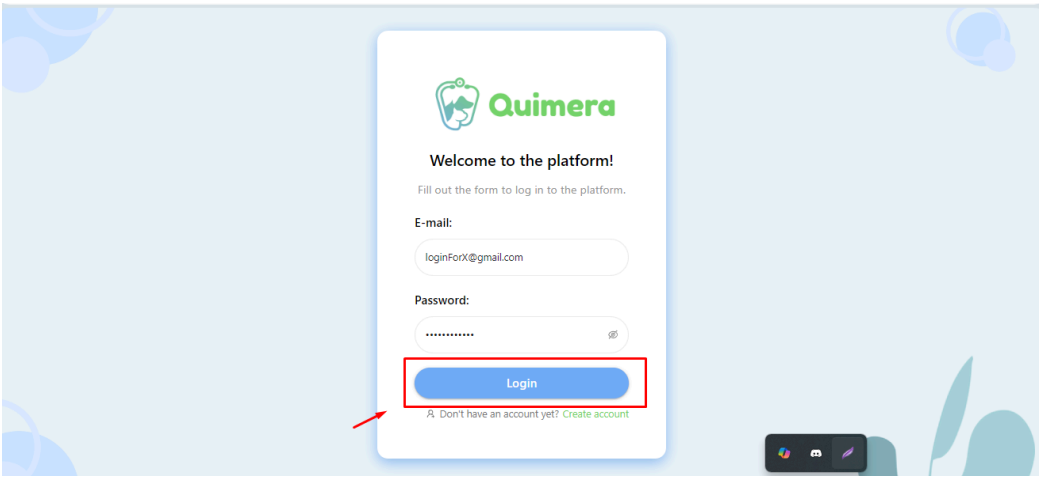
Result Tables Folders: This folder contains files with arrays extracted from Excel spreadsheets that represent graphs of body water drop. The data was divided into different files for better organization.

.env file: This is where sensitive application settings are stored, such as the database connection string. For security reasons, this file is not included in the Git repository.

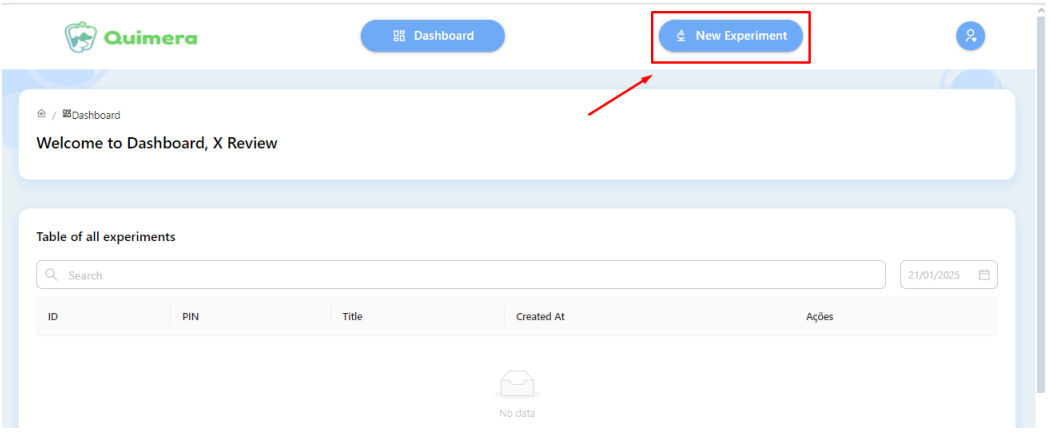
.gitignore file: Responsible for excluding files that should not be shared in the Git repository from version control, such as node_modules, yarn_lock and .env.

Step by step guide on how to use all the features

- **Teacher:**
    1. Log in with, Email: loginForX@gmail.com, Password: XisHere@2025

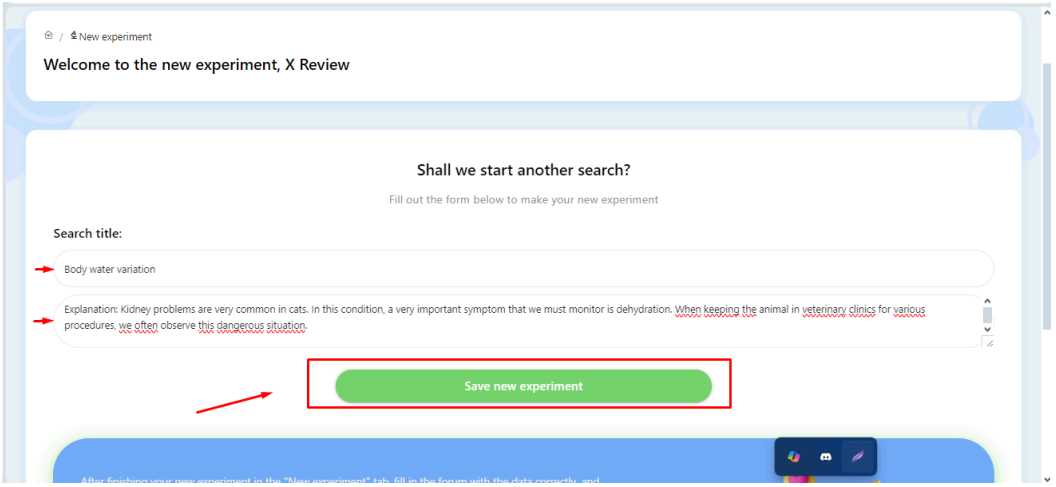


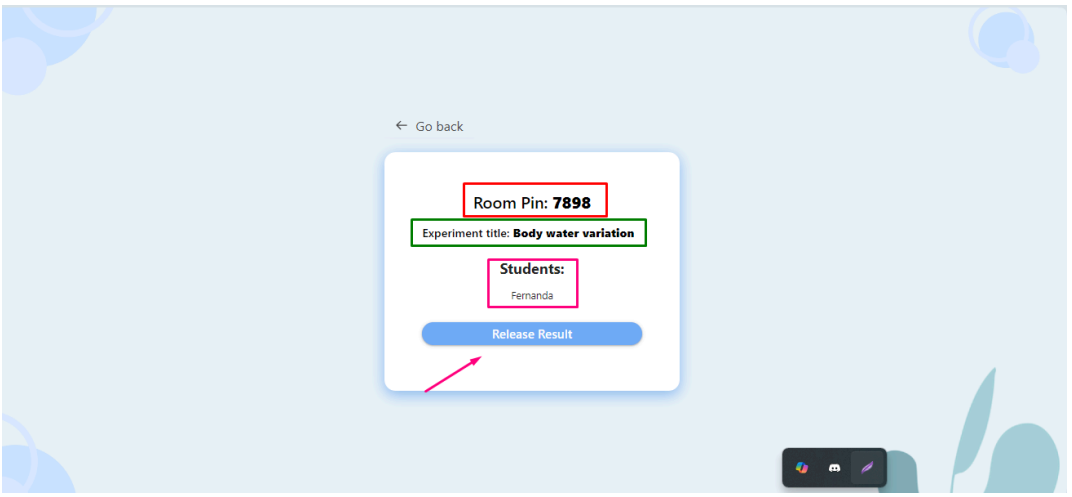
    2. Create a new experiment by clicking the "New Experiment" button in the header



    3. Just give a title and description of the experiment. For example: title, Body water variation. description, Explanation: Kidney problems are

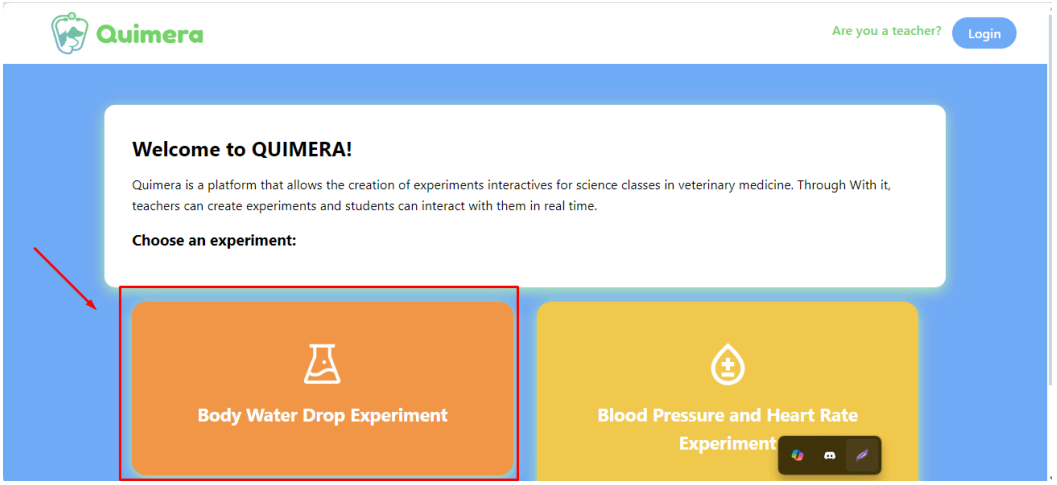
very common in cats. In this condition, a very important symptom that we must monitor is dehydration. When keeping the animal in veterinary clinics for various procedures, we often observe this dangerous situation.



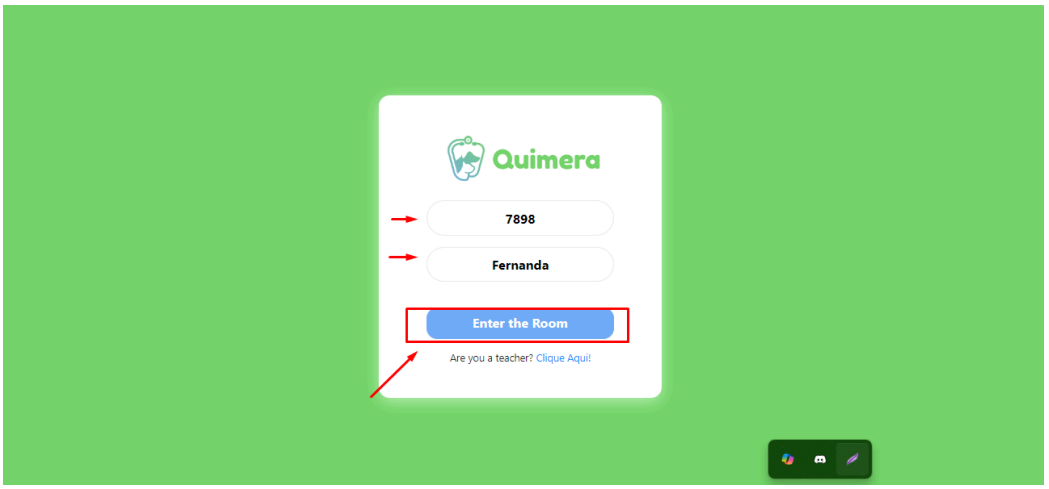
4. In the teacher's experiment room, he has access to the students' experiment room pin. In the classroom, he makes it available for students to enter. In addition, he has access to the experiment title and a list that will show all students who have entered and are entering the room.
   After the students have completed the experiment and notified the teacher, he clicks the button to release the experiment results.
5. 

- **Student**
  1. Open a new page in the same browser running localhost:3000 and Click on the card, "Body Water Drop Experiment". To log in as a student

and enter the experiment room.

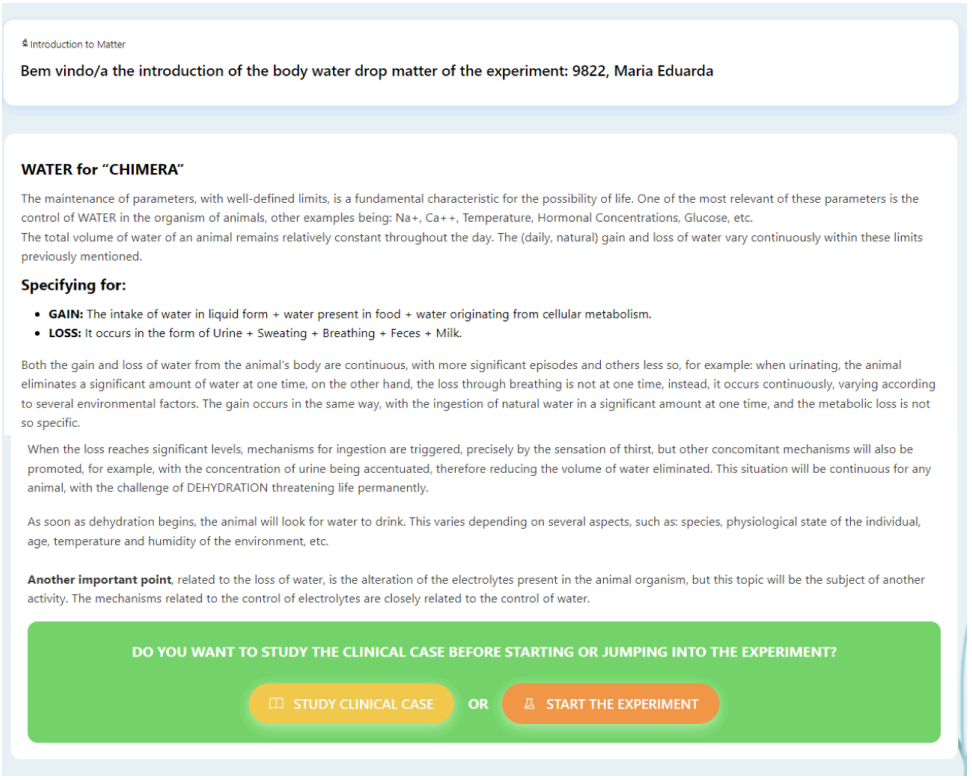


2. Place the classroom pin available on the teacher's page and the student's name. In this case, the idea was for the teacher to share the generated pin in the classroom and the students just place it and their names.

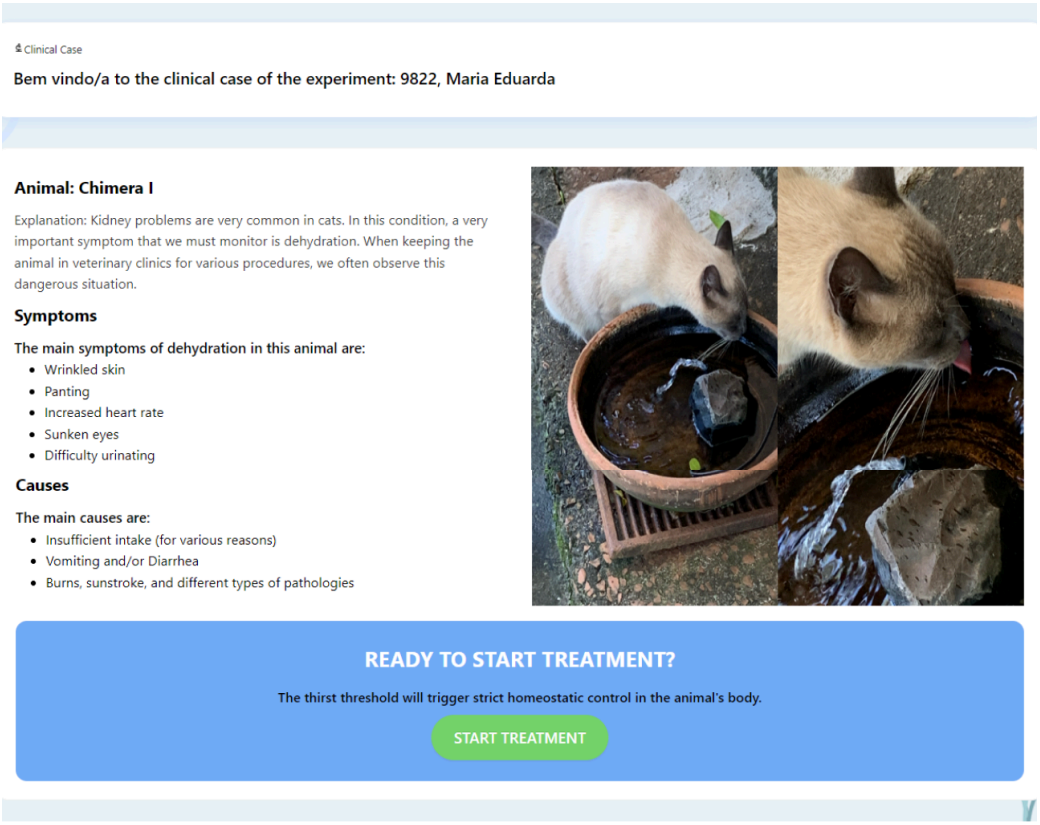


3. Loading experiment room page

4. The first page before the experiment is the introduction. On this page the student will learn about the subject of the experiment. In this case, the experiment of cats' body water falling.



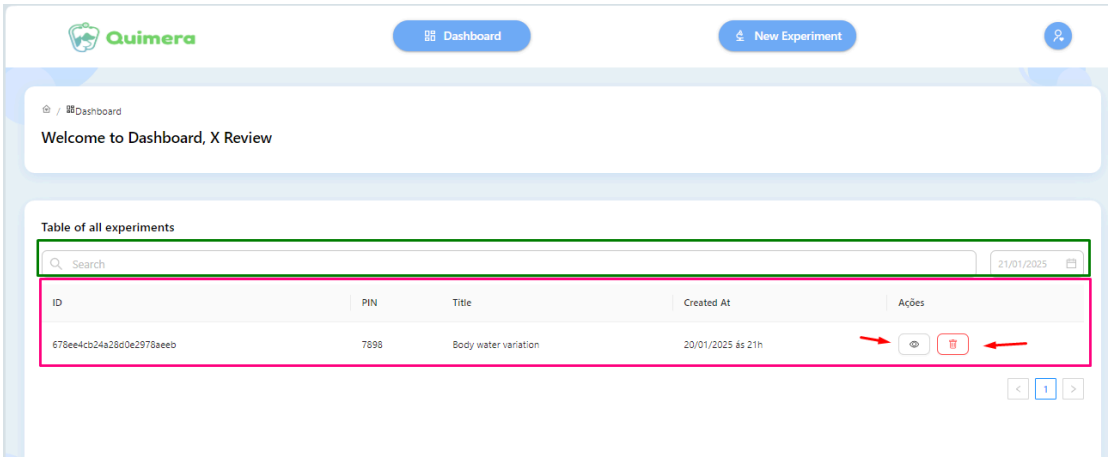
5. On the second page, you have access to the clinical case of the animal that is experiencing a loss of body water, with symptoms and causes.



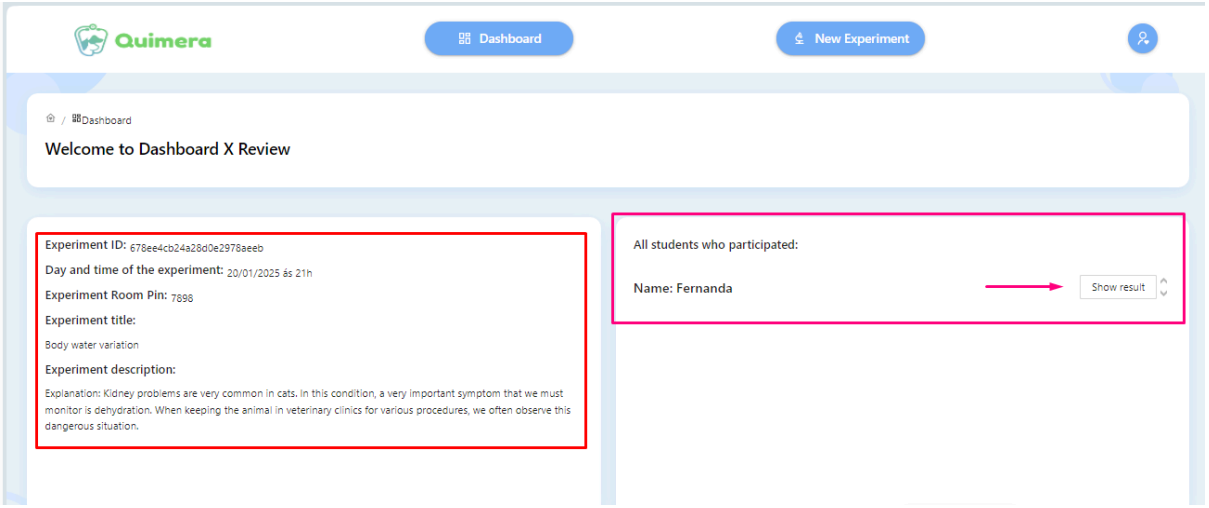
6. And finally, the experiment page where he chooses 2 hormones to correct the water loss. ADH, which corrects 20% and Hipolatome, which corrects 80%. If he chooses both, he will earn 10 points, Hipolatome 8 points and ADH 2 points. If he gets both wrong, he will earn 0 points and will not "save the animal". The animal's state is visualized by the graph, where we can see on the y-axis the time of 1 day and on the x-axis the loss that occurred.

- **Teacher:**

5. Returning to the teacher's part. In the dashboard, you can see the latest experiments performed in the table, filter by experiment and see details about it.



6. By clicking on the button with the eye icon you will see the details of the experiment. In the list of the card, "All students who participated", there is the result of all the students who participated in the experiment, just click on the "Show result" button.



7. Clicking on the "Show result" button shows the points the student got in the experiment, the graph he generated when choosing the answers and the student's name.