

# Cats and Dogs Quiz Summary

## The Idea

We decided to create a **cat** and **dog quiz** application utilizing multiples APIs.

- The user would be presented with **questions** about an animal(s).
  - Once a question was answered, a **fact** about the animal in question would be displayed.
  - At the end, a **result** would be displayed.
- 

## Quizzes

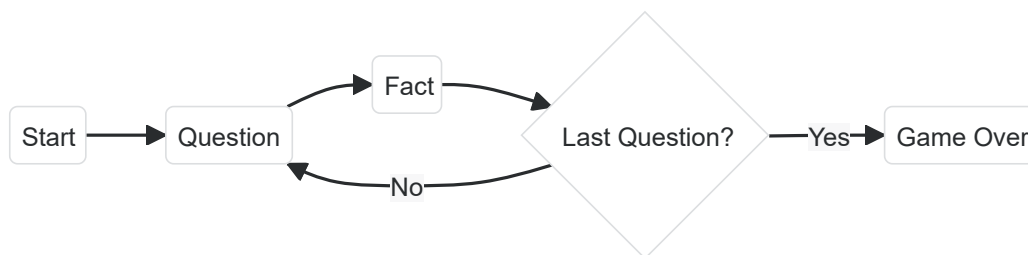
Quizzes can be separated into 2 different categories.

First being a quiz which had a **single score**. The quiz keeps track of a single score, regardless of what type of animal the question was about. At the end, the score will be **compared against the number of questions**, and a result of the quiz would be displayed. 3 quizzes were created using this category - Cat Breeds, Dog Breeds and Cat & Dog Breeds.

The other being a quiz which had **multiple scores**. The quiz keeps track of scores for cat and dog questions **individually**, and at the end the scores would be **compared against each other** to determine what **type of person** you are. A single quiz was created using this category - Cat/Dog Quiz.

---

## Game Loop



## Wed Development Process

During the **development** process, a Kanban (Trello) board was used to **manage** the project. Various tasks needing completion - features, bug fixes, styling, documentation, refactoring, were added, given a priority and taken care of in an iterative manner.

We dealt with **design**, thinking how to integrate multiple APIs together to create an engaging experience for the user. It also involved responsive design, making sure the application **looked and functioned correctly** on multiple different devices.

During **implementation**, tasks were completed **based on priority**. Higher level ones were done, followed by lesser ones. During the process, new tasks kept being discovered, mainly as new features, bug fixes, and refactoring.

The application also had to be **tested**. Each new function/change had to be tested on multiple different screen sizes. Any errors found were given a priority level based on severity, and **removed** and **minimized**.