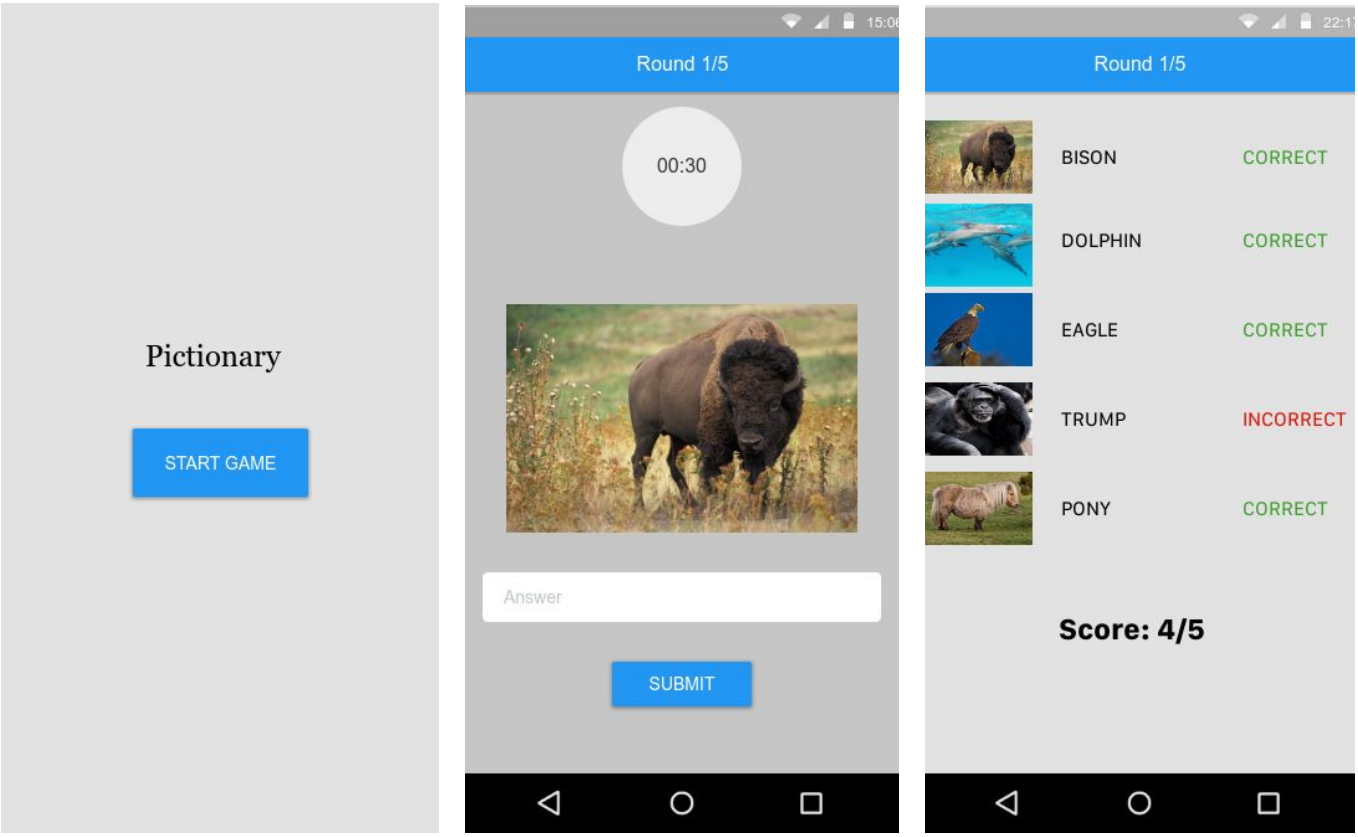
Pictionary



Given is a list of animal images and a JSON providing their attributes like difficulty and correct answer. Use these assets to make a simple game.

- When the game starts user is presented with an image of medium difficulty of level 3. If the answer is correct, the next image is presented with difficulty level increased by 1 i.e. 4 and if the answer is wrong, the next question will be of difficulty decreased by 1 i.e. 2. After every question, depending on the answer, the difficulty level of the next question is either incremented by 1 if the answer is correct or decremented by 1 if the answer is wrong.

- There is a timer running on each image screen as well. If the timer expires before the user submits the answer, it'll be treated as a wrong answer and the difficulty level will be decreased by 1 for the next image.

- The timer is 30 seconds for each screen.

- The user will be presented with 5 images max.

- If the user reaches level 1 and answers wrong, the game will be over then and there even if he/she has not attempted 5 answers. So even if this is his/her 3rd image of difficulty level 1 and the user answers incorrectly, the game is over.

- If the user reaches level 5 and answers correctly, the game continues. Next question will be of level 5 if he/she answers right and decrement to 4 if he/she answers wrongly.

- You have a pool of 5 animal images for each difficulty level, so none of the animals should be repeated.

- **an image should be selected randomly from the pool of images for the required difficulty level.**

- Calculate the number of correct answers, printing it in a toast message on completion of the game.

- Show a “Game Over” toast if he finishes the game before 5 questions (due to incorrect answers).

- **Take the user to the starting screen in both the above scenarios (completion of the game)**

*** Extension:**
(NOTE - ONLY attempt this once you’ve completely finished implementing the above requirements. Completing the bonus without meeting requirements will yield no results.)

- Build a Results screen displaying correct/incorrect answers, **printing user entry & actual answer in case of incorrect answer.** Print the difficulty level of each question, as well as sum of all correct answers.

Expectation:

1. The UI should be as shown in the illustrations.
2. Code should be **complete and correct.**
3. Overall code should be in a state to release.
4. Code should be clean, readable, testable, modular, extensible
5. The proper naming convention of the language used should be followed

Note:
JSON and image can be downloaded from **http://bit.ly/2m8YFLZ**