

LABORATORY 3

1. [20pts] Draw a **sequence diagram** of the scenario when a game tick/frame occurs starting from the `tickHandler` self-call of `FlappyBirdBrowserView`.
 - (a) Assume that:
 - i. `FlappyBirdPresenter` has already called `this.view.addTickHandler(this.handleTick)` beforehand (*as seen in its constructor*)
 - ii. No other object has called `addTickHandler`
 - iii. The game is **not yet over**
 - (b) Exclude the following:
 - i. The `window.requestAnimationFrame(this.tickHandler)` call in `tickHandler` of `FlappyBirdBrowserView`
 - ii. Any call by `FlappyBirdBrowserView` involving `this.ctx`
 - (c) Message details can be given as types, variable names, or descriptive phrases
 - (d) You are strongly encouraged to use the **Mermaid online editor** to create the sequence diagram (*click Sample Diagrams > Sequence Diagram*)

2. Feedback items

- (a) How did you find the exercise in terms of enjoyment, ease, learning, usefulness, etc.?

This was the first time that there we're no video guides which made it quite difficult to start off. The difficult part of it was understanding the data structures and how the variables interact with one another.

- (b) How long did it take you to finish each item?

It took me half a day to deal with the first collision detection and just 1 hour for the scoring.

- (c) Did you try reading through `FlappyBirdBrowserView`? If so, list down notable learnings or pain points, if any (*no need to explain in detail*)

No

- (d) How could this exercise me improved?

The tips were very helpful maybe visualize what the attributes of the bird and pipes are next time. I had trouble identifying which part of them are the attributes specifically referring to.

3. State until which part of the exercise were you able to finish:

- (a) Required item (Collision Detection, 80pts) - **Done**
- (b) Optional item (Scoring +20pts) - **Done**