Nacnas, Kevin Dave W. **CIT 3rd Year B**

Flappy Fish

1. Introduction

Flappy Fish is a 2D browser-based game developed using HTML, CSS, and JavaScript. It is inspired by the original *Flappy Bird* game concept but features a fish character ("isda" is Filipino for "fish"). The game challenges players to navigate through pipes while avoiding collisions, with features such as score tracking, high score saving, background music, and offline detection.

---

2. Technologies Used

- **Frontend:** HTML5, CSS3, JavaScript (ES6)
- **Media:** Audio (MP3), Images (GIF/PNG)
- **Browser API:** localStorage, requestAnimationFrame, EventListeners

---

3. Game Structure and Flow

3.1 HTML Structure

- : Game background container.
- <img class="isda">: Player character (fish).
- : Game start modal with start and mute options.
- : Game over modal with score display and play again button.
- <audio id="bg-music">: Background music (looping MP3).
- and : Score and high score display.

3.2 CSS Styling

The visual design is handled through external CSS (style.css), responsible for:

- Positioning and animation of game elements
- Responsive layout
- Modal designs
- Pipe and background movement visuals

3.3 JavaScript Logic

Game States:

- **Start**: Displays welcome screen.
- **Play**: Activates gameplay loop, starts gravity and pipe generation.
- **End**: Displays game over screen and saves score.

Core Variables:

- move_speed: Speed at which pipes move.
- grativy: Simulated gravity pulling the fish downward.
- isda_dy: Current vertical velocity of the fish.
- high_score: Stored in localStorage.

Main Functions:

- play(): Initializes gravity, pipe creation, and animation loop.
- apply_gravity(): Simulates gravity and updates fish's position.
- create_pipe(): Randomly generates pipe pairs with spacing.
- handleJump(): Applies an upward velocity on input.
- endGame(): Ends the game and updates high score.

Input Handlers:

- click, touchstart, and keydown events to control jump.
- click events for start, play again, and mute buttons.

Offline Detection:

Uses window.addEventListener('offline') and online to show a modal alert when internet is lost.

4. Features

Feature	Description
Dynamic Pipes	Randomly generated pipes with vertical gaps.
Gravity Simulation	Gradual acceleration downwards to simulate falling.
Jump Mechanic	Mouse click or keyboard press gives upward velocity.
Score Tracking	Player earns points by passing pipes.
Local High Score	Browser stores highest score using localStorage.
Background Music	Looping audio starts on interaction, with mute/unmute options.
Responsive UI	Works on various screen sizes and input types.
Offline Detection	Notifies user if the internet connection is lost.

5. Media Assets

Type	File Path
Fish Animation	assets/isda.gif
Background Music	sounds effect/bgmsc.mp3
Sound Effects	sounds effect/point.mp3, sounds effect/die.mp3