Design Documentation

Gill, Surinder	Hu, Joshua	Lago, Nick
1308896	1311940	1302613

November 6, 2015

Contents

Revision History

Table 1: Revision History: Proof of Concept Plan

DATE	DEVELOPER	CHANGE	REVISION
November 6, 2015	Gill, Surinder	Initial Draft	0
November 6, 2015	Hu, Joshua	Initial Draft	0
November 6, 2015	Lago, Nick	Initial Draft	0

Introduction

This document indicates the Module Interface Specifications for the implementation of 'FloppyFish'. The intent of this document is to facilitate design and maintenance of the project.
Complementary documents include the System Requirement Specifications and Module Guide.

Module Hierarchy

Table 2: Module Hierarchy

Level 1	**Level 2**	**Level 3**
Hardware-Hiding Module		
Behaviour-Hiding Module	Rendering Module	Animation Module
Software Decision Module	Initializer Module	Collision Module

MIS's of Namespace.js

`Flappy_Fish`

Name	In	Out	Exceptions
init	-	-	-
resize	-	-	-
update	-	-	-
render	-	-	-
loop	-	-	-
changeState	state	-	-

Assumptions

requestAnimFrame(input) function is defined in a previous JavaScript class so Animation Frame functions can be called in the Namespace.

State Variables

Flappy_Fish.WIDTH: int
Flappy_Fish.HEIGHT: int
Flappy_Fish.scale: int
Flappy_Fish.offset.top: int
Flappy_Fish.offset.left: int
Flappy_Fish.entities: array
Flappy_Fish.currentWidth: null
Flappy_Fish.currentHeight: null
Flappy_Fish.canvas: null
Flappy_Fish.ctx: null
Flappy_Fish.score.taps: int
Flappy_Fish.score.coins: int
Flappy_Fish.distance: int
Flappy_Fish.digits: array
Flappy_Fish.fonts: array
Flappy_Fish.RATIO: null
Flappy_Fish.game: null
Flappy_Fish.currentWidth: null
Flappy_Fish.currentHeight: null
Flappy_Fish.canvas: null
Flappy_Fish.ua: null
Flappy_Fish.android: null

Flappy_Fish.ios: null

Environment Variables

Screen: Display Output Device
Mouse: Input Device
Keyboard: Input Device
Speakers: Audio Output Device

Access Program Semantics

init: Transition: Initializes multiple game variables and sets the canvas for the desired device. This function initializes the game .
resize: Transition: Maintains the aspect ratio of the game after resizing of the browser window and across multiple platforms.
update: Transition: Update the game and restore the tapped state of the game.
loop: Transition: Iterates the updating and rendering of the game.
changeState: Input: The state of the game that it is being changed to.
changeState: Transition: The state of the game that it is being changed to.

MIS's of Main.js

`Flappy_Fish.Draw`

Name	In	Out	Exceptions
clear	-	-	-
rect	int, int, int, int, int	-	-
circle	int, int, int, int	-	-
image	image, int, int	-	-
sprite	image, int , int, int, int, int, int, int, int	-	-
semiCircler	int, int, int, int	-	-
text	string, int, int, int, int	-	-

Assumptions

No assumptions

State Variables

Environment Variables

screen: computer/device display

Access Program Semantics

Input: Each function takes in an assortment of integers used for the location of the drawing to take place. In addition to this some take in a string or image depending on what the function is drawing.
Transition: All of these functions will *only* modify the state of the environment variable screen.

`Flappy_Fish.Input`

Name	In	Out	Exceptions
set	function	-	-

Assumptions

No assumptions

State Variables

x: int
y: int
tapped: boolean

Environment Variables

Access Program Semantics

Input: This will take in data in the form of a function.
Transition: This does not modify anything. Later another function will use the data from input to modify the screen displaying the game.

`Flappy_Fish.BottomBar`

Name	In	Out	Exceptions
update	-	-	-
render	-	-	-
respawn	-	-	-

Assumptions

No assumptions

State Variables

x: int
y: int
r: int
vx: int
bg: image
name: string

Environment Variables

Screen: Display Device

Access Program Semantics

Update:
Transition: Uses the velocity varaible (vx) to update the location variable (x) of the bottom bar.
Render:
Transition: Modifies the screen, drawing the bottom bar at the locations specified.
Respawn:
Transition: Modifies the variable x which controls the location of the bottom bar on the horizontal axis.

`Flappy_Fish.Pipe`

Name	In	Out	Exceptions
update	-	-	-
render	-	-	-
respawn	-	-	-
randomIntFromInterval	int, int	int	-

Assumptions

No assumptions

State Variables

centerX: int
coin: int
w: int
h: int
vx: int
type: string

Environment Variables

Screen: Display Device

Access Program Semantics

Update:
Transition: Uses the velocity varaible (vx) to update the location variable (centerx) of the pipe.
Render:
Transition: Modifies the screen, drawing the pipe and coin (if the coin hasn't been obtained) at the locations specified. (Using draw.sprite)
Respawn:
Transition: Modifies the varaible that controls the x axis locaiton of the pipe and calls randomIntFromInterval to modify where the gap will be.
randomIntFromInterval: Takes in a min and max value both as integers.
Output: Returns a number chosen at "random" from inbetween the min and max values.

`Flappy_Fish.Bird`

Name	In	Out	Exceptions
update	-	-	-
render	-	-	-

Assumptions

No assumptions

State Variables

img: image
gravity: int

width: int
height: int
ix: int
type: string
iy: int
fr: int
vy: int
vx: int
velocity: int
play: int
jump: int
rotation: int

Environment Variables

Screen: Display device
Speakers: Devices speakers if applicable

Access Program Semantics

Update:
Transition: Updates the location and rotation of the bird sprite. Modifies environmental variable speaker if applicable.
Render:
Transition: Modifies the screen, drawing the birdat the locations specified.

`Flappy_Fish.Particle`

Name	In	Out	Exceptions
update	-	-	-
render	-	-	-

Assumptions

No assumptions

State Variables

x: int
y: int

r: int
col: int
type: string
name: string
dir: int
vx: int
vy: int
remove: boolean

Environment Variables

Screen: Display device

Access Program Semantics

Update:
Transition: adds velocity in the x (vx) and in the y (vy) to the particle. Updating its location varaibles (x,y).
Render:
Transition: Modifies environmental variable screen by using the draw function to draw a star or a circle.

`Flappy_Fish.Collides`

Name	In	Out	Exceptions
collides	`Flappy_Fish.Bird`, `Flappy_Fish.Pipe`	c1 or c2	-

Assumptions

No assumptions

State Variables

bx1: int
by1: int
bx2: int
by2: int
upx1: int
upy1: int

upx2: int
upy2: int
1px1: int
1py1: int
1px2: int
1py2: int
c1: int
c2: int

Environment Variables

Access Program Semantics

`Window.Splash`

Name	In	Out	Exceptions
init	-	-	-
update	-	-	-
render	-	-	-

Assumptions

No assumptions

State Variables

banner: image

Environment Variables

Screen: Display Device
Speaker: Device Speakers (if applicable)

Access Program Semantics

init:
Transition: Modifies Floppy Fish's varaible distance and Floppy Fish's variable storing taps and Flappy Fish's variable storing coins gathered (to an initial value of 0). Pushes the bottombar (using the BottomBar function) at an initial coordinate. Lastly modifies the environmental variable speaker by playing a sound.

update:
Transition: calls all the update functions for evey entitiy pushed. Will change the state to play if the player has tapped.
render:
Transition: modifies the screen by using the Draw funciton in Floppy Fish to draw the banner.

`Window.Play`

Name	In	Out	Exceptions
init	-	-	-
update	-	-	-
render	-	-	-

Assumptions

No assumptions

State Variables

checkCollision: boolean

Environment Variables

Screen: Display Device
Speaker: Device Speakers (if applicable)

Access Program Semantics

init:
Transition: Pusheds 3 pipes, a bird and the counter.
update:
Transition: modifies the gradient of the game.
render:
Transition: Modifies the count varaibles and draws them to the screen.

`Window.GameOver`

Name	In	Out	Exceptions
getMedal	-	string	-
getHighScore	-	`FloppyFish.score.coins`	-
init	-	-	-
update	-	-	-
render	-	-	-

Assumptions

No assumptions

State Variables

checkCollision: boolean

Environment Variables

Screen: Display Device
Speaker: Device Speakers (if applicable)

Access Program Semantics

init:
Transition: Modifies environmental variable speaker by playing sound. Updates state variables from multiple other funcitons.
update:
Transition: Modifies the value of Floppy Fish variable holding if the user has tapped.
render:
Transition: Modifies the environmental variable screen as it uses the draw function to draw images and text.
getMedal:
Output: Outputs a string containing which medal the user has won.
getHighScore:
Output: Outputs the maximum amount of coins the user has received based on the cookies the user has stored.

MIS of cookieHandler.js

`cookieHandler.js`

Name	In	Out	Exceptions
getCookie	string	string	-
setCookie	string, int, int	-	-

Assumptions

No assumptions

State Variables

name: string
ca: string
c: string
d: object
bigNum: long
expires: string

Environment Variables

path: cookie storage

Access Program Semantics

getCookie: Checks for a cookie with the name returns the cookie name as a string.
setCookie: Creates a cookie with a value and expiry date.

MIS of sound.js

`sound.js`

Name	In	Out	Exceptions
play_sounds	string	-	-

Assumptions

No assumptions

State Variables

soundJump: object
soundScore: object
soundHit: object
soundDie: object
soundSwoosh: object
sounds_max: int
sounds_length: int
audiosounds: object

Environment Variables

speaker: computer/device audio output

Access Program Semantics

play_sounds: If the audio objects n the array object are not finished load and play them.

MIS of windowSetter.js

`window.requestAnimFrame`

Name	In	Out	Exceptions
requestAnimFrame	-	function	-

Assumptions

No assumptions

State Variables

Environment Variables

screen: computer/device display

Access Program Semantics

requestAnimFrame: Requests an animation frame for various browsers and times out if no applicable animation frame is available.