Ewww!! WASABI

By TJ

6030289721 Teeti Watanatada

6030511021 Worapol Krichmaitri

2110215 Programming Methodology - Project

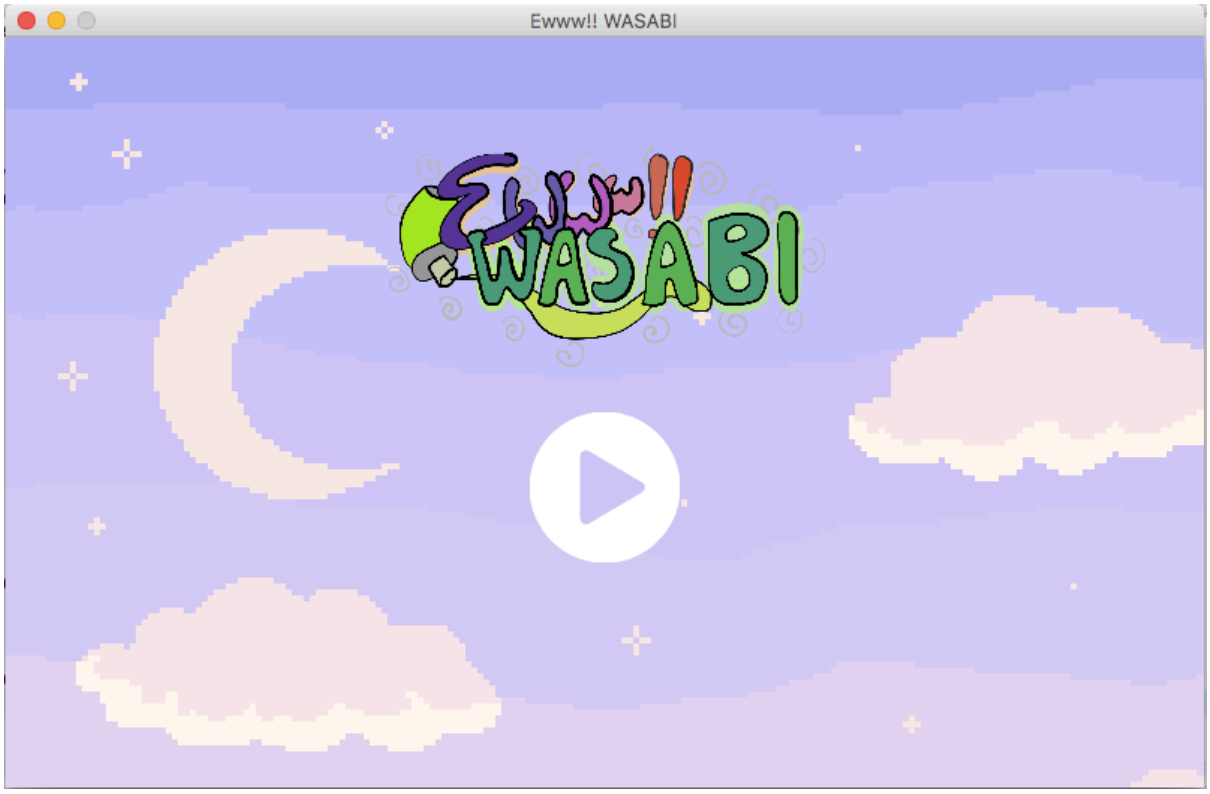
First Semester 2018

ABOUT GAME

Ewww!! WASABI is a pick sushi game. The player is a guy who doesn't like wasabi. You have to pick sushi as much as you can in 60 seconds. But be careful, there aren't only sushi appears in this game. Hehe.

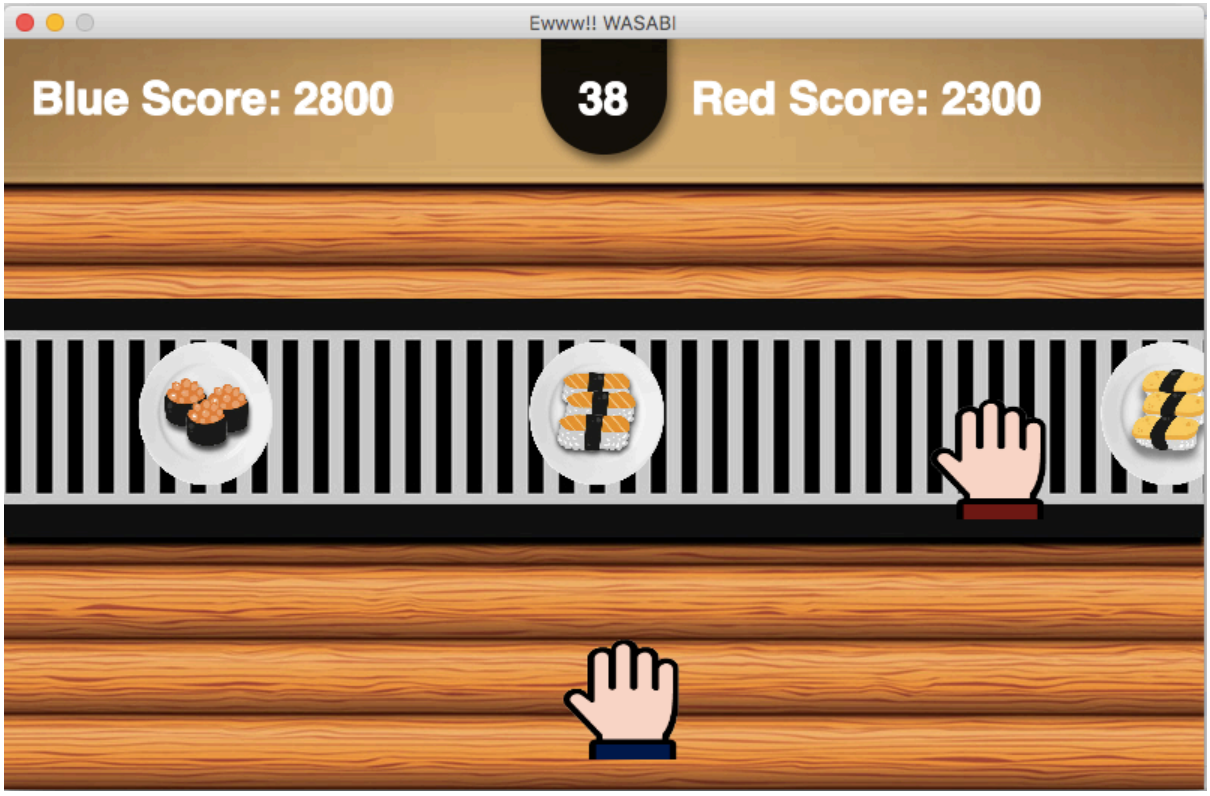
HOW TO PLAY

Main Menu



There is a play button on the center of the screen among the beautiful purple sky with sparkling stars. Click the play button to begin the game.

Game Stage



There are many sushi plates moving on the center. This game required 2 players, don't forget to bring your friend before playing. You are playing as a hand which can move up to pick sushi. Each sushi has different score. On the top of the screen, there is a score bar with a time counter in the middle. One round takes 60 seconds to play.

Controlling

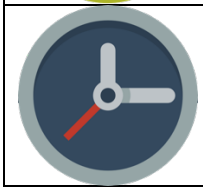
Blue team's hand use:
Key "W" to move up.
Key "A" to move left.
Key "D" to move right.

Red team's hand use:
Key "UP" to move up.
Key "LEFT" to move left.
Key "RIGHT" to move right.

Common Item

	Tuna Sushi Increase score 1,500 points when player picks.
	Salmon Sushi Increase score 1,200 points when player picks.
	Salmon Roe Sushi Increase score 1,000 points when player picks.
	Shrim Sushi Increase score 800 points when player picks.
	Tamago Sushi Increase score 500 points when player picks.
	Rotten Sushi Decrease score 3,000 points when player picks.

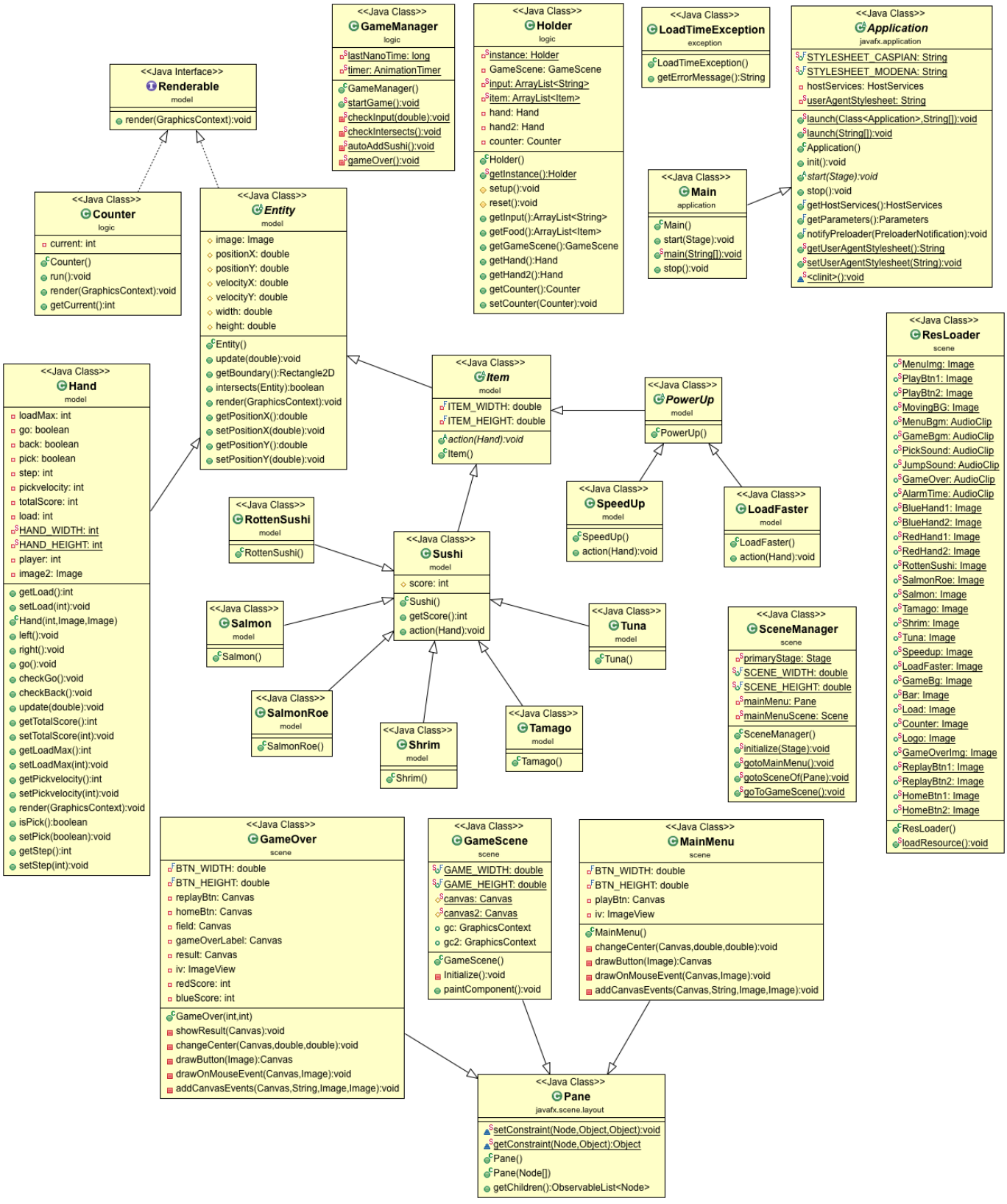
Power Up Item

	Speed Up Item Increase your hand move speed after picking this item.
	**Load Faster Item** Decrease hand reload time after picking this item.

Game Over Scene

This scene shows final score when the game ends of each players. You can try again by click the retry button or go to main menu by click the home button.

IMPLEMENTATION DETAILS



** Noted that Access Modifier Notations can be listed below*

+ (public)
 # (protected)
- (private)
static will be underlined.
abstract will be italic.

1. Package application

1.1 Class Main extends Application

1.1.1 Method

+ void start(Stage primaryStage)	Main entry point for the JavaFX applications
+ void main(String[] args)	An entry point of the application.

2. Package exception

2.1 Class LoadTimeException extends Exception

2.1.1 Method

+ String getErrorMessage( )	Return message “Can’t pick when your hand is reloading.”

3. Package Logic

3.1 Class Counter extends Thread implements Renderable

3.1.1 Field

- int current	The current time in the game start from 60.

3.1.2 Method

+ void run( )	Countdown game time every 1 second.
+ void render( )	Draw current countdown time.
+ int getCurrent( )	Getter of current time.

3.2 Class GameManager

3.2.1 Field

- long lastNanoTime	The last time that use for compute elapseTime.
- AnimationTimer timer	An AnimationTimer object use for handle all updates in game. - Increase hand's load value. - Render all component (paintComponent) - Check input from key board. - Add Items. - Check an intersect status. - Update all items. - Check time up and go to gameOver scene.

3.2.2 Method

+ void startGame( )	Start game - Setup objects in Holder. - Start Counter. - Start Animationtimer.
- void checkInput( )	Control two hands by check an input from key board.
- void checkIntersects( )	Check intersect status and get action from item.

- void autoAddItem( )	Random generate item and add them to item Arraylist.
- void gameOver( )	Stop AnimationTimer, reset objects in holder, and goto gameover scene.

3.3 Class Holder

3.3.1 Field

- Holder instance	A singleton of holder.
- GameScene GameScene	A GameScene object.
- ArrayList<String> input	ArrayList of input in the game.
- ArrayList<Item> item	ArrayList of item in the game.
- Hand hand	A hand, A player in the game.
- Hand hand2	A hand, Another player in the game.
- Counter counter	A time counter.

3.3.2 Constructor

+ Holder( )	Initialize an input ArrayList, item ArrayList.

3.3.3 Method

+ Holder getInstance( )	Getter of instance.
# void setup( )	Initialize GameScene, hand of two players, time counter.
# void reset( )	Reset all field.
+ ArrayList<String> getInput( )	Getter of input ArrayList.
+ ArrayList<Item> getFood( )	Getter of item ArrayList.
+ GameScene getGameScene( )	Getter of GameScene.
+ Hand getHand( )	Getter of hand.
+ Hand getHand2( )	Getter of hand2.
+ Counter getCounter( )	Getter of counter.
+ void setCounter(Counter counter)	Setter of counter.

4. Package Model

4.1 *Class Entity* implements Renderable

4.1.1 Field

# Image image	Object image in the game.
# double positionX	X Position of object in the game.
# double positionY	Y Position of object in the game.
# double velocityX	Velocity of object in X axis.
# double velocityY	Velocity of object in Y axis.
# double width	Object width.
# double height	Object height.

4.1.2 Method

+ void update(double time)	Update object's position by given time.
+ Rectangle2D getBoundary( )	Get the bounds of an object.
+ Boolean intersects(Entity e)	Check that two object are intersects.
+ void render(GraphicsContext gc)	Draw image by given GraphicsContext.
+ double getPositionX( )	Get X Position of object in the game.
+ void setPositionX(double positionX)	Setter of positionX.
+ double getPositionY( )	Get Y Position of object in the game.
+ void setPositionY(double positionY)	Setter of positionY.

4.2 Class Hand

4.2.1 Field

- Image image2	An image when the hand is picking.
- int HAND_WIDTH	Width of hand.
- int HAND_HEIGHT	Height of hand.
- int load	An int object that collect hand's load value.

	If load value > loadMax value, then hand can pull (go).
- int loadMax	An int object that indicate load maximum value. If load value > loadMax value, then hand can pull (go).
- int step	An int object that indicate a temporary velocity(Y) of Hand.
- boolean go	A boolean object that indicate hand go status (move up).
- boolean back	A boolean object that indicate hand back status (move down).
- boolean pick	A boolean object that indicate hand pick status (intersect).
- int pickVelocity	An int object that indicate a temporary velocity(X) of Hand.
- int totalScore	Total score of each player's hand.
- int player	An int object that indicate hand or hand2.

4.2.2 Constructor

+ Hand(int player, Image image, Image image2)	Set hand field. - Image, Image2 - Set player (hand, hand2) - loadMax = 400. - pickvelocity = 300 - load = 400 - step = 5 - (For hand) positionX = 250 / positionY = 400 (For hand2) positionX = 500 / positionY = 400 - velocityX = 0 - velocityY = 0 - totalScore = 0 - go = false - back = false - pick = false

4.2.3 Method

+ void left( )	Move left by set positionX = positionX-step.(Can't move when the hand at leftmost , go or back).
+ void right( )	Move right by set positionX = position+step.(Can't move when the hand at rightmost , go or back).
+ void go( )	Move up by set velocityY = - pickVelocity (Can't move when the hand's load value less than loadMax or back).
+ void checkGo( )	Check go status if positonY less than 200 set the hand's field. -velocityY = pickvelocity(back). - go = false. - back = true. - pick = false.
+ void checkBack( )	Check back status if positonY more than 400 set the hand's field. - velocityY = 0(stop) - go = false - back = false - positionY = 400 - pick = false
+ void update(double time)	(Override) Check go and back status and update hand's position.
+ int getLoad( )	Getter of load.
+ void setLoad(int load)	Setter of load.
+ int getTotalScore( )	Getter of totalScore.
+ void setTotalScore(int score)	Setter of tatalScore.
+ int getLoadMax( )	Getter of loadMax.
+ void setLoadMax(int loadMax)	Setter of loadMax.
+ int getPickVelocity( )	Getter of pickVelocity.
+ void setPickVelocity(int pickvelocity)	Setter of pickVelocity.
+ boolean isPick( )	Check the hand's pick status.
+ void setPick(boolean pick)	Setter of pick.
+ int getStep( )	Getter of step.

+void setStep(int step)	Setter of step.
+ void render(GraphicsContext gc)	Draw the hand - If pick is true, then draw hand by image2. Else draw by image. - If load less than loadMax draw Resloader.Load(load progress) - Draw total score for each hands(hand,hand2).

4.3 *Class Item*

4.3.1 Field

- final double ITEM_WIDTH	Item width.
- final double ITEM_HEIGHT	Item height.

4.3.2 Constructor

+ Item( )	Set position, velocity, width and height of an item.

4.3.3 Method

+ *void action(Hand hand)*	Abstract Method.

4.4 Class LoadFaster extends PowerUp

4.4.1 Constructor

+ void LoadFaster( )	Set load faster item's image.

4.4.2 Method

+ void action(Hand hand)	Decrease loadMax of hand.

4.5 *Class PowerUp* extends Item

No field, constructor or method in this class

4.6 Interface Renderable

4.6.1 Method

+ void render(GraphicsContext gc)	(Interface) Draw an object.

4.7 Class RottenSushi extends Sushi

4.7.1 Constructor

+ RottenSushi( )	Set rotten sushi's image and score.

4.8 Class Salmon extends Sushi

4.8.1 Constructor

+ Salmon( )	Set salmon sushi's image and score.

4.9 Class SalmonRoe extends Sushi

4.9.1 Constructor

+ SalmonRoe( )	Set salmon roe sushi's image and score.

4.10 Class Shrim extends Sushi

4.10.1 Constructor

+ Shrim( )	Set shrim sushi's image and score.

4.11 Class SpeedUp extends PowerUp

4.11.1 Constructor

+ SpeedUp( )	Set speed up item's image.

4.11.2 Method

+ void action(Hand hand)	Increase hand velocity and step.

4.12 Class Sushi extends Item

4.12.1 Field

# int score	Score of each type of sushi.

4.12.2 Method

+ int getScore( )	Get sushi score.
+ void action (Hand hand)	Increase hand totalScore by sushi score.

4.13 Class Tamago extends Sushi

4.13.1 Constructor

+ Tamago( )	Set tamago sushi's image and score.

4.14 Class Tuna extends Sushi

4.14.1 Constructor

+ Tuna( )	Set tuna sushi's image and score.

5. Package scene

5.1 Class GameOver extends Pane

5.1.1 Field

- final double BTN_WIDTH	Width of the button.
- final double BTN_HEIGHT	Height of the button.
- ImageView iv	Set background as a gif.
- Canvas replayBtn	Canvas of replay button.
- Canvas homeBtn	Canvas of home button.
- Canvas field	Canvas of text field.
- Canvas gameOverLabel	Canvas of game over label.
- Canvas result	Canvas of result field.
- int redScore	Final score of red hand's player.

- int blueScore	Final score of blue hand's player.

5.1.2 Constructor

+ GameOver(int score1, int score2)	Initialize red hand's score and blue hand 's score as given values Initialize all buttons and Draw all canvases .

5.1.3 Method

- void showResult (Canvas canvas)	Show the result of the game includes red score and blue score.
- void changeCenter(Canvas canvas, double x, double y)	Change the center of canvas.
- Canvas drawButton(Image img)	Draw a button as given image.
- void drawOnMouseEvent(Canvas canvas, Image img)	Change the image of the button when mouse enter.
- void addCanvasEvents(Canvas canvas, String buttonName, Image img1, Image img2)	- If you click on replay button, it will go to GameScene. - If you click on home button, it will go to MainMenuScene. - If your mouse enter replay button or home button, it will change the button image. - If your mouse exit replay button or home button, it will change back the button image.

5.2 Class GameScene

5.2.1 Field

- final double GAME_WIDTH	Width of the game stage scene.
- final double GAME_HEIGHT	Height of the game stage scene.
# Canvas canvas	A canvas for background.
# Canvas canvas2	Canvas of objects in game stage.
+ GraphicsContext gc	Draw game background and bar.
+ GraphicsContext gc2	Draw objects (e.g. sushi).

5.2.2 Constructor

+ GameScene( )	Initialize the game scene.

5.2.3 Method

- void Initialize( )	Draw game stage background and bar.
+ void paintComponeent( )	Draw and clear objects in the game.

5.3 Class MainMenu extends Pane

5.3.1 Field

- final double BTN_WIDTH	Width of button.
- final double BTN_HEIGHT	Height of button.
- Canvas playBtn	Canvas of play button.
- ImageView iv	ImageView of main menu background.

5.3.2 Constructor

+ MainMenu( )	Initialize all buttons and draw all canvases.

5.3.3 Method

- void changeCenter(Canvas canvas,double x, double y)	Change the center of canvas.
- Canvas drawButton(Image img)	Draw a button as given image.
- void drawOnMouseEvent (Canvas canvas, Image img)	Change the image of the button when mouse enter.
- void addCanvasEvents (Canvas canvas, String buttonName, Image img1, Image img2)	- If you click on play button, it will go to GameScene. - If your mouse enter play button, it will change the button image. - If your mouse exit play button, it will change back the button image.

5.4 Class ResLoader

5.4.1 Field

+ AudioClip MenuBgm	Sound of menu background music.
+ AudioClip GameBgm	Sound of game background music.
+ AudioClip PickSound	Sound of PickSound.
+ AudioClip JumpSound	Sound of JumpSound.
+ AudioClip GameOver	Sound of GameOver.
+ AudioClip AlarmTime	Sound of AlarmTime.
+ Image MenuImg	Image of menu background.
+ Image PlayBtn1	Image of play button when mouse enter.
+ Image PlayBtn2	Image of play button when mouse exit.
+ Image MovineBG	Image of menu moving background.
+ Image BlueHand1	Image of blue team's hand
+ Image BlueHand2	Image of blue team's hand when pick an object.
+ Image RedHand1	Image of red team's hand.
+ Image RedHand2	Image of red team's hand when pick an object.
+ Image RottenSushi	Image of rotten sushi.
+ Image Salmon	Image of salmon sushi.
+ Image SalmonRoe	Image of salmon roe sushi.
+ Image Tamago	Image of tamago.
+ Image Shrim	Image of shrim.
+ Image Tuna	Image of tuna.
+ Image SpeedUp	Image of speed up item.
+ Image LoadFaster	Image of load faster item.
+ Image GameBg	Image of game background.
+ Image Bar	Image of top bar in game scene.
+ Image Load	Image of load symbol in the game.
+ Image Counter	Image of counter.
+ Image Logo	Image of logo of the game in main menu.
+ Image GameOverImg	Image of game over background.
+ Image ReplayBtn1	Image of replay button when mouse enter.
+ Image ReplayBtn2	Image of replay button when mouse exit.
+ Image HomeBtn1	Image of home button when mouse enter.
+ Image HomeBtn2	Image of home button when mouse exit.

5.4.2 Method

+ void loadResource( )	Load all resources in res folder.

5.5 Class SceneManager

5.5.1 Field

- Stage primaryStage	Primary stage.
+ final double SCENE_WIDTH	Width of scene.
+ final double SCENE_HEIGHT	Height of scene.
- Pane mainMenu	Pane of main menu.
- Scene mainMenuScene	Scene of main menu.

5.5.2 Method

+ void initialize(Stage stage)	Set primaryStage to given stage. Set primaryStage resizable to be false Show primaryStage.
+ void gotoMainMenu( )	Set scene to mainMenuScene.
+ void gotoSceneOf(Pane pane)	Set scene by given Pane.
+ void gotoGameScene( )	Start game and set scene to GameScene.