SpaceCraft Documentation

Created by

Anon Ongsakul 6231374421

2110215 Programming Methodology

Semester 2 Year 2019

Chulalongkorn University

SpaceCraft .

Introduction.

SpaceCraft is inspired by the Sci-fi classic game “Space Invaders”, the arcade game in 1978. The story of the game is when the universe is chaos, the world is facing with villains and asteroids from other solar system. The SpaceCraft forces have duty to against the attack of villains and asteroids. Player get a role to a captain of a SpaceCraft unit. The objective of this game is survival for a long time as possible.

Instruction.

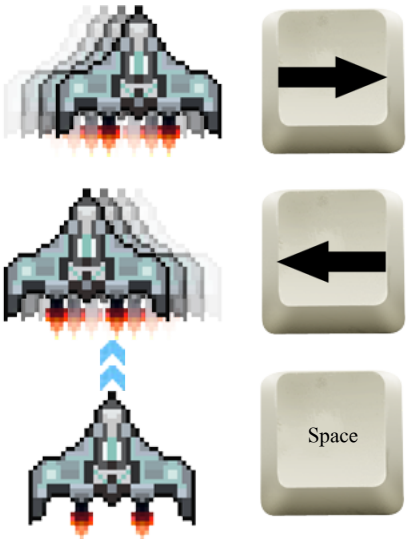
To protect our world. Player have to control the SpaceCraft unit by keyboard input. The duties of the captain of the SpaceCraft unit are enemies destruction and avoiding collision with villains and asteroids. The player can take 3 action for this game, moving to the left ,moving to the right and firing the bullet. The keyboard inputs for taking actions are depended on player setting. There are buff objects to enhance the damage of the SpaceCraft unit and heal Hp of the SpaceCraft unit. When the SpaceCraft unit collects buff object, The object emit the aura to protect the SpaceCraft unit.

As default, the keyboard input for taking action as follows :

- “Right arrow” for moving the SpaceCraft unit to the right.
- “Left arrow” for moving the SpaceCraft unit to the left.
- “Space bar” for control the SpaceCraft unit to shoot .

While player is playing the game, the player can quit to main menu scene or restart the game by pressing Escape button.

Example of Controller.

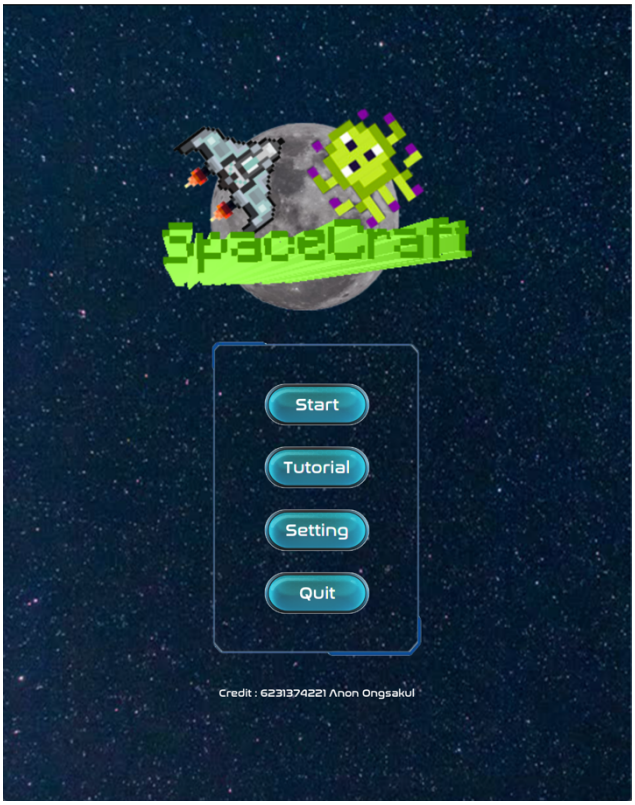


Press right arrow key to move the unit to the right.

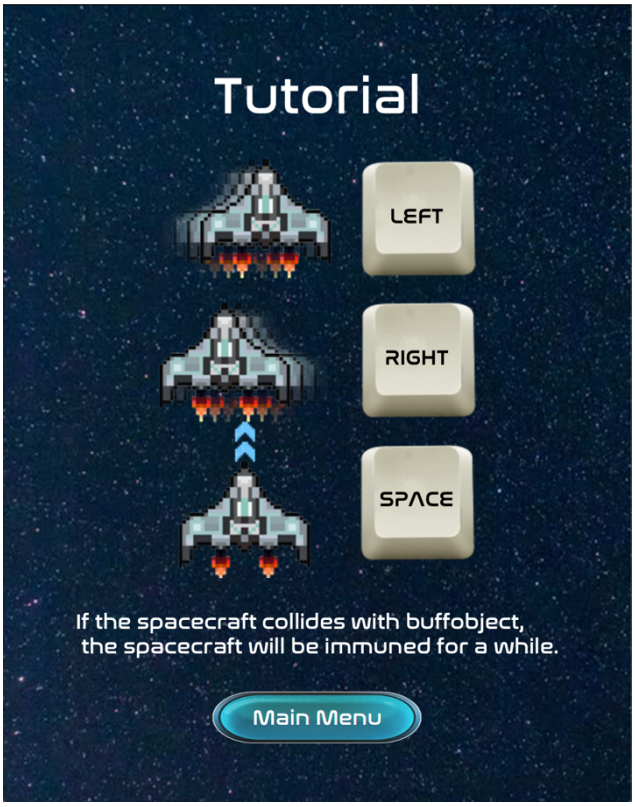
Press left arrow key to move the unit to the left.

Press space bar key to fire the bullet.

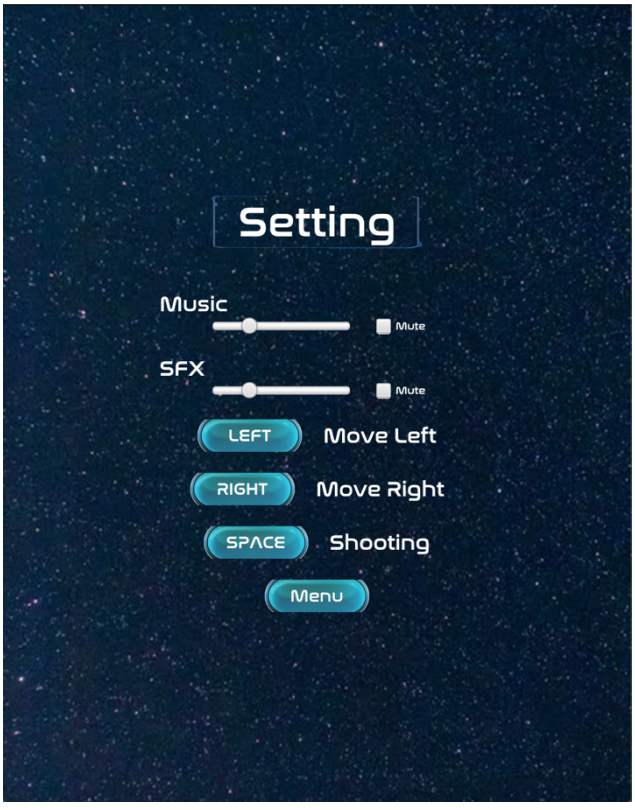
Main menu scene.



Tutorial scene.

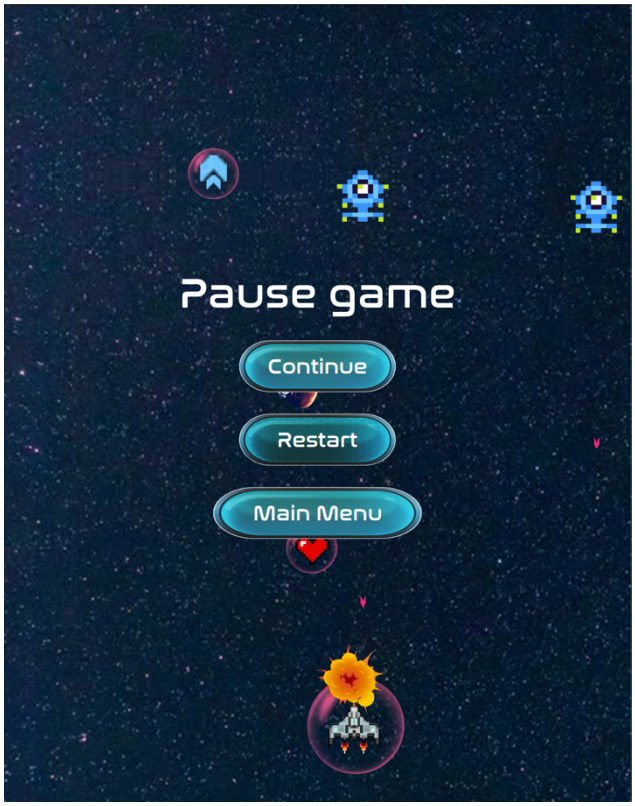


Setting scene.

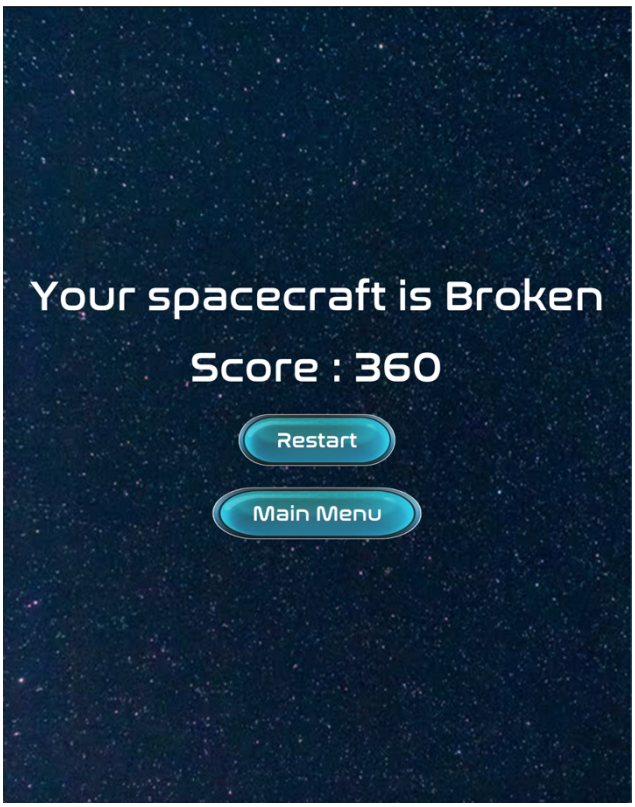


Game scene.

Paused game scene.



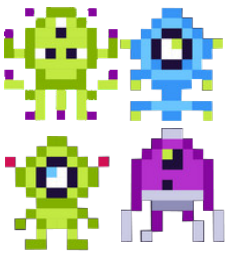
Game over scene.



Entities explanation.

SpaceCraft (Player) has Max HP 100 points , damage 20 points and fire 30 damage bullet. SpaceCraft will be destroyed when it's HP lower than 0.

Player's bullet has 30 damage and is destroyed when collide with villains and asteroids.



Villains have Max HP 100 points , damage 20 points and fire 25 damage bullet. Villains will be destroyed when it's HP lower than 0.

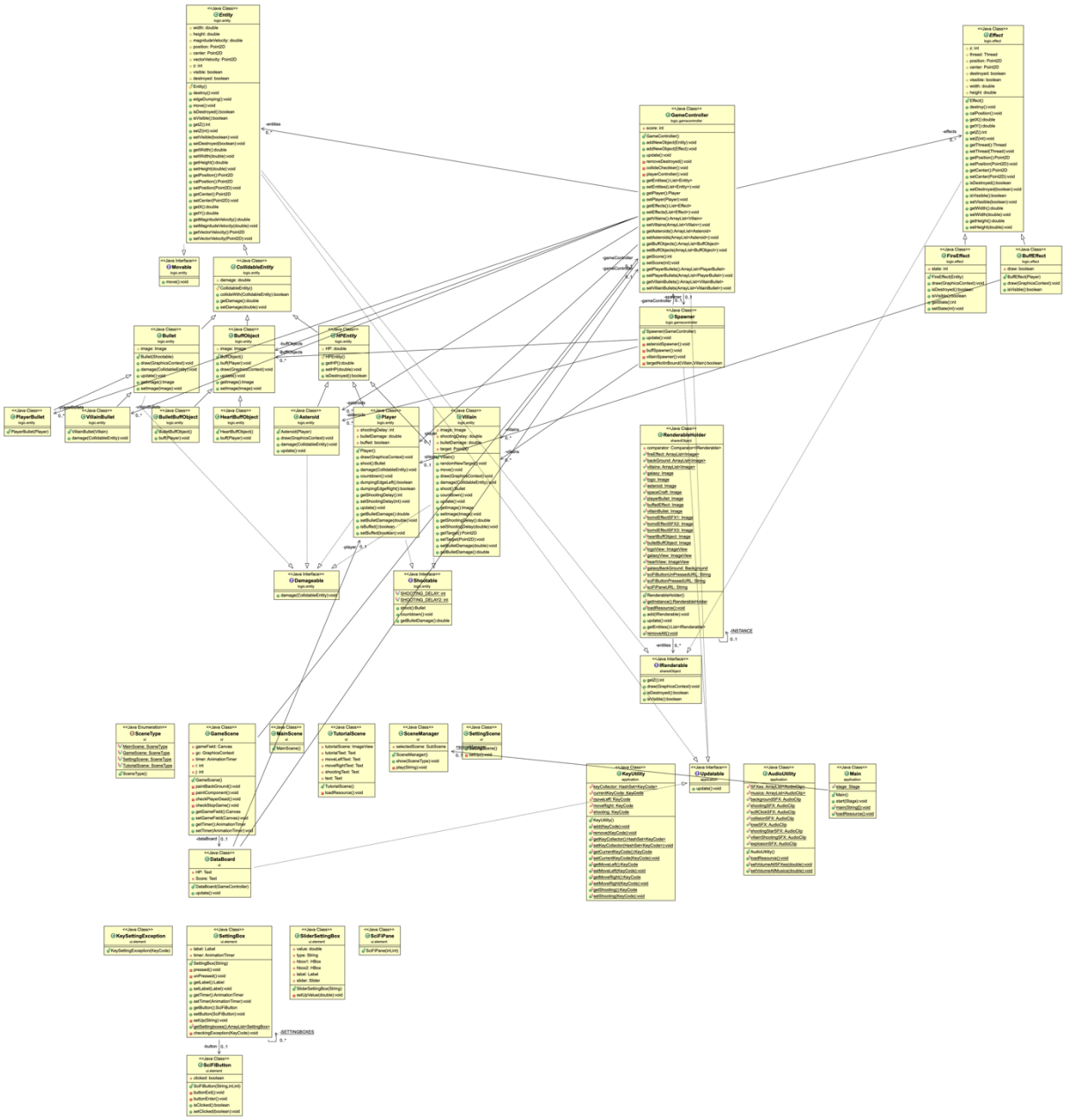
Villain’s bullet has 30 damage and will be destroyed when collide with SpaceCraft(Player).

Asteroids are have Max HP 100 points and damage 40 points. Asteroid will be destroyed when it’s HP is lower than 0 or collide with the SpaceCraft (Player).

Heart buff object is a object that can heal the SpaceCraft (Player) to Max HP and create aura to protect the SpaceCraft (Player) for a while. It will be destroyed when collide with the SpaceCraft

Bullet buff object is a object that can in It will be destroyed when collide with the SpaceCraft crease the SpaceCraft (Player)’s bullet damage with ten percent and create aura to protect the SpaceCraft (Player) for a while. It will be destroyed when collide with the SpaceCraft

UML diagram.



Notation

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

+ (public), # (protected), - (private), underlined (static), ALL_CAPS (final)

1 Package sharedObject ***Copy from the instructor's code***

1.1 Interface Irenderable

1.1.1 Method

+int getZ()	Return value of Z of the object.
+void draw(GraphicsContext gc)	Draw on canvas's graphic context.
+boolean isDestroyed()	Return boolean of Destroyed of the object.
+boolean isVisible()	Return boolean of Visibleof the object.

1.2 Class RenderableHolder

1.2.1 Field

-RenderableHolder INSTANCE	The only one RenderableHolder of the class.
-List<IRenderable> entities	List of drawable objects.
-Comparator<IRenderable> comparator	The comparator of entities.
+Images in the game.	

1.2.2 Constructor

+RenderableHolder()	Initialize entities and comparator.

1.2.3 Method

+void loadResource()	Load images of the game.
+void add()	Add object to the list of drawable objects and sort them by value of Z.
+void removeAll()	Remove all objects in instance.
+getter setter	

1.2.4 Block

Call loadResource()	

2 Package logic.entity

2.1 Interface Damageable

2.1.1 Method

+abstract void damage(CollidableEntity other)	The ability to damage HPEntity.

2.2 Interface Movable

2.2.1 Method

+abstract void move()	The ability to move of objects.

2.3 Interface Shootable

2.3.1 Field

+int SHOOTING_DELAY	The ability to move of objects.
+int SHOOTING_DELAY2	The ability to move of objects.

2.3.2 Method

+abstract Bullet shoot()	The Shootable objects can shoot and return Bullet to collect.
+ abstract void countdown()	The Shootable objects can countdown.
+ abstract double getBulletDamge()	Return damage of shootable object's bullet .

2.4 Abstract Class Entity implements IRenderable,Movable

2.4.1 Field

#double width	Width of entity.

#double height	Height of entity.
#double magnitudeVelocity	Magnitude of entity's velocity.
#Point2D position	Position of entity in coordinate system.
#Point2D center	Center position of entity in coordinate system.
#Point2D vectorVelocity	A unit vector in coordinate system.
#int z	The order to drawable entity.
#boolean visible	The status of entity to be observe.
#boolean destroyed	The status of entity to be not exist.

2.4.2 Constructor

#Entity()	The initialize method of entity and set status of entity to visible and not destroy.

2.4.3 Method

+void edgeBumping()	The checker method for entity collide with the edge of the game scene and destroy them.
+void move()	The method to change position of entity with the calculation between vectorVelocity and magnitudeVelocity.
+void move()	The method to change position of entity with the calculation between vectorVelocity and magnitudeVelocity.
+void Point2D calPosition()	The method calculate position from the center.
+getters and setters methods of all fields.	

2.5 Abstract Class CollidableEntity extends Entity

2.5.1 Field

#double damage	Damage of CollidableEntity.

2.5.2 Constructor

#CollidableEntity()	The initialize method of collidable entity.

2.5.3 Method

+void collideWith(CollidableEntity other)	The checker method for interaction between the collidable entity and other collidable entity.
+A getter and a setter for a field.	

2.6 Abstract Class HPEntity extends ColliableEntity

2.6.1 Field

#double HP	Health point of HPEntity.

2.6.2 Constructor

# HPEntity ()	The initialize method of HPEntity entity.

2.6.3 Method

+Boolean isDestroyed()	The HP checker method. When HP of the HPEntity lower than 0, it will be destroyed and can't be observable.
+A getter and a setter for a field.	

2.7 Class Player extends HPEntity

2.7.1 Field

-int shootingDelay	Shooting delay of player.
-double bulletDamge	The damage of player's bullets.
-boolean buffed	The status of player buff.

2.7.2 Constructor

+Player()	The initialize method of player. It set width and height to 50, buffed to false, magnitude of velocity to 5, zero vector of velocity, position in (X,Y) coordinate system to (275,675), shootingDelay to 0, it's HP to 100, z to one, damage to 20 and bulletdamage to 30. It calculate the center of player boundary by other method.

2.7.3 Method

+void draw(GraphicsContext gc)	This method draw player image in canvas' GraphicContext by it's position.
+Bullet shoot()	This method set shooting to Shootable.SHOOTING_DELAY and return bullet for storage in list of bullet.
+void damage(ColliableEntity other)	This method reduce player's HP and destroy or reduce other's HP depends on the class of other.
+void countdown()	This method reduce shootingDelay.
+boolean dumpingEdgeRight()	The checker method for right edge collision.
+boolean dumpingEdgeLeft()	The checker method for left edge collision.
+void update()	This method call move method.
+getters and setters methods of all fields.	

2.8 Class Villain extends HPEntity

2.8.1 Field

-double shootingDelay	Shooting delay of player.
-double bulletDamge	The damage of player's bullets.
-Image image	The image for identical type of villain.
-Point2D target	The target of the position when it is spawned.

2.8.2 Constructor

+ Villain ()	The initialize method of villain. It set width and height to 50, magnitude of velocity to 0, a unit vector of velocity to down position, random shootingDelay, it's HP to 100, z to 1, damage to 20 and bulletdamage to 25. It random the target to land and random it's image.

2.8.3 Method

+void randomNewTarget()	This method call for random target point and set the position follow the randomized target.
+void draw(GraphicsContext gc)	This method draw villain's image on graphic context of canvas.
+void move()	This method calculate magnitude of velocity by decreased-exponential function of the position and call superclass's move method.
+void damage(CollidableEntity other)	This method reduce villain's HP and destroy or reduce other's HP depends on the class of other.
+Bullet shoot()	This method fire when villain is ready to shoot, return bullet to storage and set shooting delay to Shootable.SHOOTING_DELAY2.
+void countdown()	This method reduce villain's shootingDelay
+void update()	This method call move and countdown method.

+getters and setters methods of all fields.	

2.9 Class Asteroid extends HPEntity

2.9.1 Constructor

+ Asteroid ()	The initialize method of asteroid. It set width and height to 40, magnitude of velocity to 2, it's HP to 100, z to two, damage to 40. It random the position on the upper edge and set the direction of a unit velocity vector by aim to the player.

2.9.2 Method

+void draw(GraphicsContext gc)	This method draw asteroid image on graphic context of canvas.
+void damage(CollidableEntity other)	This method is called when the asteroid collide with other CollidableEntity to reduce asteroid's HP. If the other is player, it will be destroy instantly or the player is buffed it can't damage the player. If the other is other class, other will be destroyed or reduced other's HP.
+void update()	This method call move and edgeDumping method.

2.10 Class Bullet extends ColliableEntity

2.10.1 Field

-Image image	The image of bullet.

2.10.2 Constructor

+ Bullet (Shootable owner)	The initialize method of bullet. It set width to 5, set height to 10,set the center ,damage of bullet by the value of owner and set z to zero. This constructor call position calculation method.

2.10.3 Method

+void draw(GraphicsContext gc)	This method draw bullet's image on graphic context of canvas.
+void damage(CollidableEntity other)	This method reduce other's HP and destroy the bullet.
+void update()	This method call move and edgeDumping methods.
+A getter and setter methods of a field.	

2.11 Class PlayerBullet extends ColliableEntity

2.11.1 Constructor

+ PlayerBullet (Player owner)	The initialize method of player's bullet. It call the constructor method of super class. It set magnitude of velocity to 10, image to RenderableHolder.playerBullet and direction to upper edge.

2.12 Class VillainBullet extends ColliableEntity

2.12.1 Constructor

+ VillainBullet (Villain owner)	The initialize method of player's bullet. It call the constructor method of super class. It set magnitude of velocity to 9, image to RenderableHolder.villainBullet and direction to lower edge.

2.12.2 Method

+void damage(CollidableEntity other)	This method reduce other's HP and destroy the villain bullet. The villain bullet can't damage the buffed player.

2.13 Class BuffObject extends ColliableEntity

2.13.1 Field

-Image image	The image of buff object.

2.13.2 Constructor

+ BuffObject()	The initialize method of buff object. It set width and height to 50, set magnitude of velocity to three, set z to two, damage to zero, vector of velocity to the lower direction. This constructor random position.

2.13.3 Method

+void draw(GraphicsContext gc)	This method draw buff object's image on graphic context of canvas.
+void buff(Player player)	This method call destroy and player's buff status to true.
+void update()	This method call move and edgeDumping methods.
+A getter and setter methods of a field.	

2.14 Class HeartBuffObject extends BuffObject

2.14.1 Constructor

+ HeartBuffObject()	The initialize method of buff object. It call super class constructor and set image to RenderableHolder.heartBuffObject.

2.14.2 Method

+void buff(Player player)	This method call super class buff and set player's HP to 100.

2.15 Class BulletBuffObject extends BuffObject

2.15.1 Constructor

+ HeartBuffObject()	The initialize method of buff object. It call super class constructor and set image to RenderableHolder.bulletBuffObject.

2.15.2 Method

+void buff(Player player)	This method call super class buff and increase player's bullet game 10%.

3 Package logic.effect

3.1 Abstract Class Effect implements IRenderable

3.1.1 Field

#int z	The order to drawable entity.
#Thread thread	The thread of effect.
#Point2D postion	The position of effect.
#Point2D center	The center of effect.
#boolean visible	The status of effect to be observe.
#boolean destroyed	The status of effect to be not exist.

#double width	Width of entity.
#double height	Height of entity.

3.1.2 Constructor

+ Effect()	The initialize method of effect. It set z to two, destroyed to false, visible to true.

3.1.3 Method

+void destroy()	This method set the destroyed to true and set visible to false.
+void calPosition()	This method calculate position by effect's center.
+getters and setters methods of fields.	

3.2 Class FireEffect extends Effect

3.2.1 Field

-int state	The state of the image that show on screen.

3.2.2 Constructor

+ FireEffect(Entity entity)	The constructor of fire effect. It set width and height to 100, get center of entity to the effect's center and initialize the thread to run the state. This constructor call the position calculator method and run it's thread.

3.2.3 Method

+void draw(GraphicsContext gc)	This method draw the fire effect image in n state.
+getters and setters methods of fields.	

3.3 Class BuffEffeect extends Effect

3.3.1 Field

-Player player	The player is followed by the buff effect.
-boolean draw	The status for drawing the image of this effect.

3.3.2 Constructor

+ BuffEffect(Player player)	The constructor of buff effect or aura. It set width and height to 100, get field player to player and draw to false and initialize the thread for blink the object . This constructor run it's thread.

3.3.3 Method

+void draw(GraphicsContext gc)	This method draw the buff effect on canvas in the same position with the player. It calculate the position before buff effect is drawed.
+getters and setters methods of fields.	

4 Package logic.gamecontroller

4.1 Class GameController implement Updatable

4.1.1 Field

-List<Entity> entities	The list of the Entities on the game.
-List<Effect> effects	The list of the Effects on the game.

-Spawner spawner	The spawner to spawn the villains and asteroids.
-int score	The player's score.
-Player player	The only one player.
-ArrayList<PlayerBullet> playerBullets	The player's bullets collector in the game.
-ArrayList<VillainBullet> villainBullets	The villain's bullets collector in the game.
-ArrayList<Villain> villains	The villains collector in the game.
-ArrayList<Asteroid> asteroids	The asteroids collector in the game.
-ArrayList<BuffObject> buffObjects	The BuffObject collector in the game.

4.1.2 Constructor

+ GameController()	The constructor of game controller in the game. It set score to zero, initialize all of the field and collect player to appropriate collector by call addNewObject method.

4.1.3 Method

+void addNewObject(Entity entity)	This method add entity to entity collector and drawable. It categorize the entity to collector by entity's subclass.
+void addNewObject(Effect effect)	This method add effect to effect collector and drawable.
+void update()	This method call the spawner to generate villains and asteroids, call player's controller method, call update of all entities in the collector, order villains to shoot, check the collision checker method and remove all objects that were destroyed.

-void removeDestroyed()	This method remove the destroyed objects in every collector in the field and show the effect if the object is villain or asteroid.
-void collideChecker()	This method check collision between buff objects and player, asteroids and player, villain bullets and player, player bullets and villain or asteroid.
-void playerController()	This method receive order from user input. It move spacecraft to left,right when user press arrow left,right key or the key that have been set. It order spacecraft to shoot right when user press spacebar key or the key that have been set and play shooting SFX.
+getters and setters methods of fields.	

4.2 Class Spawner implements Updatable

4.2.1 Field

-ArrayList<Villain> villains	The villains collector in the game.
-ArrayList<Asteroid> asteroids	The asteroids collector in the game.
-ArrayList<BuffObject> buffObjects	The BuffObject collector in the game.
-GameController gameController	The gameController in the game.

4.2.2 Constructor

+ Spawner (GameController gameController)	The constructor of Spawner in the game. It set all field to be consistent with the game controller.

4.2.3 Method

+void update()	This method call local spawner.

-void asteroidSpawner()	This method spawn asteroid if the random more than threshold and don't more than capacity.
-void buffSpawner()	This method spawn buff object if the random more than threshold and don't than capacity.
-void villainSpawner()	This method spawn villain if the random more than threshold, don't than capacity and the villain's target boundary doesn't overlay others villain's target boundary.

5 Package ui

5.1 Enum SceneType

MainScene	The variable for main scene.
GameScene	The variable for game scene.
SettingScene	The variable for setting scene.
TutorialScene	The variable for tutorial scene.

5.2 Class SceneManager

5.2.1 Field

-SubScene selectedScene	The current scene that show on the stage.

5.2.2 Constructor

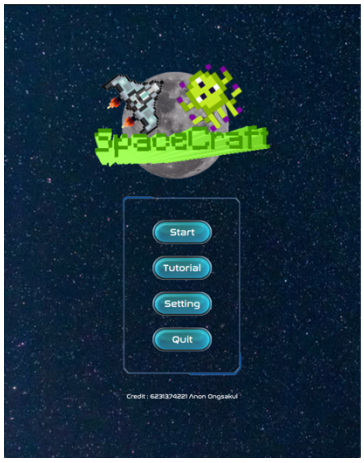
+ SceneManager()	The constructor of scene manager in the game. It add background on it and initialize selected scene to show main scene.

5.2.3 Method

+void show()	This method remove old selected scene and show new selected scene by calling play method to move selected scene.
-void play()	This method move selected scene from the right edge.

5.3 Class MainScene

5.3.1 Constructor

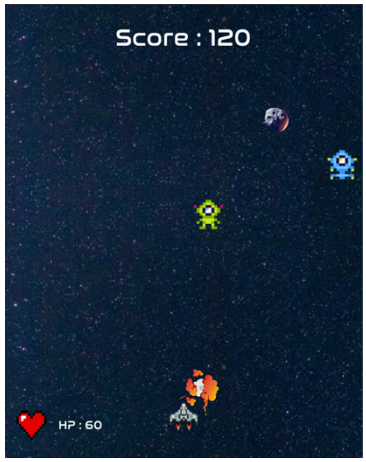
+ MainScene ()	The constructor of main scene in the game. It play music sound and provide the element of main scene follow the picture and set size of game scene. 

5.4 Class GameScene extends StackPane

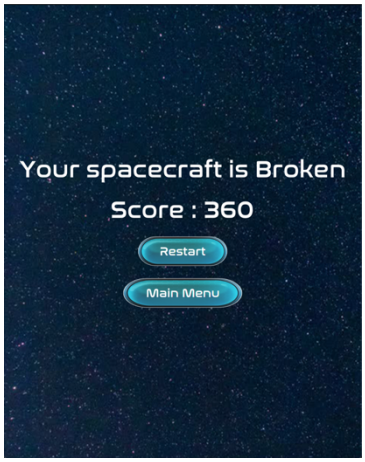
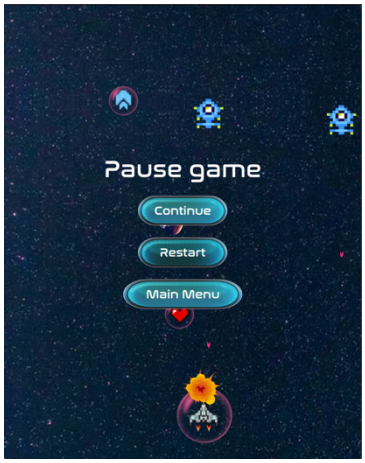
5.4.1 Field

-GameController gameController	The game controller of the game.
-Canvas gameField	The performer on the stage.
-AnimationTimer timer	The game runner.
-DataBoard dataBoard.	The board that show player's Hp and score.
-int i,j	

5.4.2 Constructor

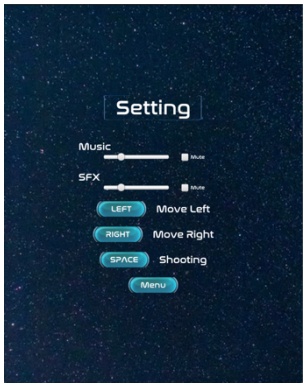
+ GameScene()	The constructor of game scene in the game. In the timer, it call to player dead and stop game checker, update method of Updatable and istance of RenderableHolder and paint the component on the canvas. It initialize all of the fields and provide element of game scene follow the picture and set size of game scene. 

5.4.3 Method

-void paintBackGround()	This method paint background like it move with infinity loop.
-void paintComponent()	This method draw IRenderable.
-void checkPlayerDead()	This method is the player death checker. When the player death, it change the element of game scene follow the picture. 
-void checkStopGame()	This method is the pause checker. When the game is paused, it change the element of game scene follow the picture 
+getters and setters methods of fields.	

5.5 Class SettingScene extends VBox

5.5.1 Constructor

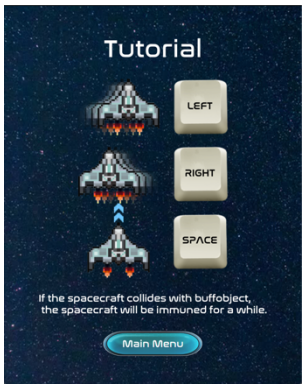
+ SettingScene()	The constructor of setting scene in the game. It provide element of setting scene follow the picture by call setUp method and set size of setting scene. 

5.5.2 Method

-void setUp()	This method set up element.

5.6 Class TutorialScene

5.6.1 Constructor

+ TutorialScene()	The constructor of tutorial scene in the game. It call resource loader method. It provide element of tutorial scene follow the picture method and set size of tutotial scene. 

5.6.2 Method

-void loadResource()	This method load resource for tutorial scene.

5.7 Class DataBoard extends Pane implements Updatable

5.7.1 Field

-Text HP	The text that appear player's Hp.
-Text Score	The text that appear score.
-Player player	The player to get HP.
-GameController gameController	The game controller to get score.

5.7.2 Constructor

+ DataBoard ()	The constructor of data board in the game. It provide element data board as appear on the GameScene and set size of data board.

5.7.3 Method

+void update()	This method update HP and Score.

6 Package ui.element

6.1 Class ScifiButton extends Button

6.1.1 Field

-boolean clicked	The clicked status of scifi button.

6.1.2 Constructor

+ ScifiButton (String title,int width.int height)	The constructor of scifi button in the game. It provide element by set style method, set size of scifi button by

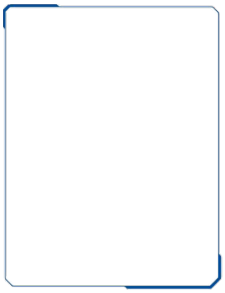
	the parameter and add event handler for mouse enter and mouse exit.

6.1.3 Method

-void buttonExit()	This method to set style the button when mouse is not over ther button. 
-void buttonEnter()	This method to set style the button when mouse is over ther button. 
+A getter and a setter methods of a field.	

6.2 Class ScifiPane extends BorderPane

6.2.1 Constructor

+ ScifiPane (int width.int height)	The constructor of scifi pane in the game. It provide element by set style method and set size of scifi pane by the parameter. 

6.3 Class SettingBox exyends HBox

6.3.1 Field

-ArrayList<SettingBox> **SETTINGBOXES**	The only one setting boxes collector of the game.
-Label label	The label of the setting box.
-AnimationTimer timer	The runner while setting box is clicked.
-SciFiButton button	The button to press.

6.3.2 Constructor

+ SettingBox (String string)	The constructor of setting box in the game. It provide element by set style method, add event handler and call setup method.

6.3.3 Method

-void pressed()	This method start the timer, set style for the label and unpressed other box in SETTINGBOXES.
-void unPressed()	This method stop the timer, set style for the label and unpressed other box in SETTINGBOXES.
-void setUp(String string)	This method is called for set up timer handler and button that depend on parameter.
-void checkingException(KeyCode keycode) throws KeySetttingException	This method is the checker for the key that set for the action is allow.
+getters and setters methods of all fields.	

6.3.4 Block

add three setting boxes to the collector	

6.4 Class SliderSettingBox extends VBox

6.4.1 Field

-double value	The value of the slider.
-String type	The title for slider setting box.
-Hbox hbox1	
-Hbox hbox2	
-Label label	The label of slider setting box.
-Slider slider	The slider to change value.

6.4.2 Constructor

+SliderSettingBox (String string)	The constructor of slider setting box in the game for SFX or Music setting. It provide element by set style method, add event handler and checkbox for mute.

6.4.3 Method

-void setUpValue(double value)	This method set the volume of SFX or Music by the parameter value.

6.5 Class KeySettingException extends Exception

6.5.1 Constructor

+ KeySettingException (KeyCode keycode)	The constructor of key setting exception in the game. It will show alert scene when the checker throw it.

7 Package application

7.1 Class Main

7.1.1 Field

-Stage stage	The stage of the game.
-SceneManager sceneManager	The scene manager for access from another class.

7.1.2 Method

+void start(Stage stage) throws Exception	This method is the game starter.
+void main(String[] args)	Main method.
+void loadResource()	Font loader.

7.1.3 Block

call loadResource()	

7.2 Class AudioUtility

7.2.1 Field

-ArrayList<AudioClip> SFXes	The SFXes collector.
-ArrayList<AudioClip> musics	The musics collector.
-AudioClips	

7.2.2 Method

-void loadResource()	This method is sound loader. It categorize the loaded sound to the appropriate collector.
+void setVolumeAllSFXes(double value)	This method set volume of all audio clip in SFXes.
+void setVolumeAllMusics(double value)	This method set volume of all audio clip in Musics.

7.2.3 Block

Call loadResource()	

7.3 Class KeyUtility

7.3.1 Field

-HashSet<KeyCode> keyCollector	The keycode collector.
-KeyCode currentKeyCode	The key that is pressed.
-KeyCode moveLeft	The key for move player to the left.
-KeyCode moveRight	The key for move player to the right.
-KeyCode shootind	The key for order player to shoot.

7.3.2 Method

+void add(KeyCode keycode)	This method add key code to the collector.
+void remove(KeyCode keycode)	This method remove key code to the collector.
+getters and setters methods of all fields.	

7.4 Interface Updatable

7.4.1 Method

+abstract void update()	The updatable can update.