SE 3XA3: Module Internal Specification
SnakeGame Project

Team L03G09
Qiang Gao (gaoq20)
Zhiwei Li (liz342)
Longwei Ye (yel16)

March 18, 2022

Contents

1 Module Hierarchy

Level 1	**Level 2**
Hardware-Hiding Module	
Behaviour-Hiding Module	View Module BGM Control Module Inputs Checking Module Game Control Module
Software Decision Module	SnakeGame Module History Module Snake Color Module

Table 1: **Module Hierarchy**

2 MIS of BGM Control Module

2.1 Interface Syntax

2.1.1 Exported Access Programs

Name	In	Out	Exceptions
BgmControl	-	-	-
SoundChange	-	-	-

2.2 Interface Semantics

2.2.1 State Variables

music: object
currentButton: object
sound: object
span: object
music: object

2.2.2 Environmental Variables

N/A

2.2.3 Assumptions

Variables should be set before trying to access them

2.2.4 Access Program Semantics

BgmControl():
Input: None
Transition: The BGM will be paused or played by clicking the button.
Output: None
Exception: None

SoundChange():
Input: None
Transition: The volume of BGM will be turned up or down by sliding the slider.
Output: None
Exception: None

3 MIS of Inputs Checking Module

3.1 Interface Syntax

3.1.1 Exported Access Programs

Name	In	Out	Exceptions
checkColorHex	-	Boolean	-
checkUserName	-	Boolean	-

3.2 Interface Semantics

3.2.1 State Variables

color_HEX: object txtUserName: object

3.2.2 Environmental Variables

N/A

3.2.3 Assumptions

Variables should be set before trying to access them

3.2.4 Access Program Semantics

checkColorHex():
Input: None

Output: True if the length is valid, false if it is not.

Exception: None

checkUserName():
Input: None

Output: True if the length is valid, false if it is not.

Exception: None

4 MIS of Game Control Module

4.1 Interface Syntax

4.1.1 Exported Access Programs

Name	In	Out	Exceptions
onCountChange	integer, String	-	Invalid Input
onGamePause	boolean	.mp4	-
onGameOver	integer	-	-
btnStart.onclick	event	-	-
btnPasue.onclick	event	-	-
updateSnake	-	updateSnake	-

4.2 Interface Semantics

4.2.1 State Variables

btnStart: object btnPasue: object snaColor: object gameHistory: String[] gameSnake: SnakeGame

4.2.2 Environmental Variables

N/A

4.2.3 Assumptions

All variables are set when the object is constructed.

4.2.4 Access Program Semantics

onCountChange():

Input: integer, Scores users got

Transition: Scores changing when user playing the game

Exception: None

onGamePause():

Input: boolean, status

Transition: play and pause the game

Exception: None

onGameOver():

Input: integer, scores

Transition: Display "game over" and scores

Exception: None

btnStart.onclick(event):

Input: click

Transition: start the game

Exception: None

btnPasue.onclick(event):

Input: click

Transition: pause the game

Exception: None

updateSnake():

Input: None

Transition: Update the status of game

Output: updateSnake

Exception: None

5 MIS of SnakeGame Module

5.1 Interface Syntax

5.1.1 Exported Access Programs

Name	In	Out	Exceptions
SnakeGame	object, object	SnakeGame	-
initSnake	-	-	-
initScense	-	-	-
genFood	-	-	-
genSpeeder	-	-	-
eatFood	object	boolean	-
eatSpeeder	object	boolean	-
gameOver	-	boolean	-
snakeMove	-	-	-
changeSpeed	-	-	-
handleKeyInput	object	-	-
initGame	-	-	-
triggerEvent	object,object	-	-
runGame	-	-	-
pauseGame	-	-	-
changeGameStatus	-	-	-
startGame	-	-	-

5.2 Interface Semantics

5.2.1 State Variables

gameScense: object
graphic: object
count: integer
itemCount: integer
itemvalid: boolean
snake: object
curFood: object
curSpeeder: object
runId: Integer
isMoved: boolean
gameStatus: boolean

curDirection: integer
size: integer
rowCount: integer
colCount: integer
snakeColor: object
foodColor: object
speederColor: object
scenseColor: object
directionKey: double
pauseKey: double
levelCount: integer
curSpeed: double
onCountChange: object
onGamePause: object
onGameOver: object

5.2.2 Environmental Variables

N/A

5.2.3 Assumptions

All variables are set when the object is constructed.

5.2.4 Access Program Semantics

SnakeGame(gameScenseId, gameConfigObj):
Input: The id for the current game and combined variables used for a game
Transition: None
Output: Construct a SnakeGame
Exception: None

initSnake():
Input: None
Transition: store the initial position of a snake in game
Exception: None

initScense():
Input: None
Transition: print the initial background of the game
Exception: None

genFood():
Input: None
Transition: print a common food in a random position

Exception: None

genSpeeder():
Input: None
Transition: print a speeder in a random position
Exception: None

eatFood(snakeHead):
Input: snakeHead, an object
Output: True if a common food is eaten by the snake, false otherwise
Exception: None

eatSpeeder(snakeHead):
Input: snakeHead, an object
Output: True if a speeder is eaten by the snake, false otherwise
Exception: None

gameOver():
Input: None
Output: true if the game is ended, false otherwise
Exception: None

snakeMove():
Input: None
Transition: Take keyboard input to update one position shift, check if any food or speeder is eaten, and if speed is needed to change.
Exception: None

changeSpeed():
Input: None
Transition: refresh speed based on current score.
Exception: None

handleKeyInput(key):
Input: key on keyboard, an object
Transition: Change moving direction based on key.
Exception: None

initGame():
Input: None
Transition: initialize the game
Exception: None

triggerEvent(callback,argument):
Input: callback,argument

Transition: template the end game, pause game events.
Exception: None

runGame():
Input: None
Transition: continue the game or end the game.
Exception: None

pauseGame():
Input: None
Transition: pause the game based on the trigger runId.
Exception: None

changeGameStatus():
Input: None
Transition: change the game status pause or continue, based on the trigger runId
Exception: None

startGame():
Input: None
Transition: do the initial part of the game, and set gameStatus
Exception: None

6 MIS of History Module

6.1 Interface Syntax

6.1.1 Exported Access Programs

Name	**In**	**Out**	**Exceptions**
displayHistory	-	-	-
pushHistory	Integer	-	-

6.2 Interface Semantics

6.2.1 State Variables

playerName: object
playerScore: Integer
toWrite: object
gameHistory: object

6.2.2 Environmental Variables

N/A

6.2.3 Assumptions

All variables are set when the object is constructed.

6.2.4 Access Program Semantics

displayHistory():
Input: None
Transition: Display the history of the game.
Output: None
Exception - None

pushHistory(count):
Input: Scores user gets
Transition: Record scores in gameHistory
Output: None
Exception - None

7 MIS of Snake Colors Module

7.1 Interface Syntax

7.1.1 Exported Access Programs

Name	In	Out	Exceptions
snaColorR.onclick	event	-	-
snaColorG.onclick	event	-	-
snaColorB.onclick	event	-	-
snaColorDefault.onclick	event	-	-
snaColorSet.onclick	event	-	-

7.2 Interface Semantics

7.2.1 State Variables

N/A

7.2.2 Environmental Variables

N/A

7.2.3 Assumptions

None

7.2.4 Access Program Semantics

snaColorR.onclick():
Input: event

Transition: Red color

Output: None

Exception: None

snaColorG.onclick():
Input: event

Transition: Green color

Output: None

Exception: None

snaColorB.onclick():
Input: event

Transition: Blue color

Output: None

Exception: None

snaColorDefault.onclick():
Input: event

Transition: Default white color

Output: None

Exception: None

snaColorSet.onclick():
Input: event

Transition: Change color pf snake

Output: None

Exception: None

8 Side Notes

The MIS document consists of two files:

- This MIS.pdf file
- Automated generated document using open-source generator *JSDoc*, with reference link: https://jsdoc.app. The generated html file is located at src/REV_0/js/out folder. Please refer for details.

9 Revision History

Date	**Version**	**Notes**
2022/3/15	1.0	MIS for Snake Color Module
2022/3/16	1.1	MIS for BGM Control Module & Game History Module
2022/3/17	1.2	MIS for Inputs Checking Module
2022/3/18	1.3	MIS for Game Control Module
2022/3/18	1.4	Revise for MIS document

Table 2: **Revision History**