Module Interface Specification for Bomber Clone Application

Software Engineering 3XA3: Project

Group 12 - The A Team
Gabriel Lopez De Leon, 1310514
Ren-David Dimen, 1222679
Jay Nguyen, 1327828

06 November 2015

Table of Contents

1 Revision History

Revision #	Revision Date	Description of Change	Author
2	Nov 05 2015	Complete MIS	Gabriel Lopez de Leon
1	Nov 05 2015	Create MIS Outline	Gabriel Lopez de Leon

2 Introduction

The following document is for the Module Interface Specifications for the various implemented modules in the Bomber Clone application. It is intended to aid in the software development process by making navigation through the program, with regards to design and maintenance, easier. Note that this document may reference the SRS and the Module Guide.

3 Module Hierarchy

The following table is taken directly from the Module Guide for this project.

Level 1	**Level 2**
Hardware-Hiding Module	
Behaviour-Hiding Module	Screen Module Sprite Module SpriteSheet Module Keyboard Module Player Module
Software Decision Module	CollisionTest Module

Table 1: Module Hierarchy

4 MIS of Screen Module

4.1 Exported Access Programs

Name	**In**	**Out**	**Exceptions**
Screen	int, int	-	-
renderPlayer	int, int, Sprite	-	-

4.2 Interface Semantics

4.2.1 State Variables

width : int
height : int

pixels : array of int
tiles : array of int

4.2.2 Environment Variables

4.2.3 Assumption

Screen will be called in the BomberGame constructor which will be initialized in the main class of the game. Screen will be used to set the display for the game when the program is run.

4.2.4 Access Program Semantics

Screen: is used to initialize the size of the screen (output display) and then sets its color to black. It takes in two parameters width and height which is used set the screen size.

renderPlayer: takes input of two integers to set the size of the player and a Sprite which will be what the player looks like. This function is used to draw the player on to the screen.

5 MIS of Sprite Module

5.1 Exported Access Programs

Name	In	Out	Exceptions
Sprite	int, int, int, SpriteSheet	-	-

5.2 Interface Semantics

5.2.1 State Variables

SIZE : int
x : int
y : int

pixels : array of int

5.2.2 Environment Variables

5.2.3 Assumption

5.2.4 Access Program Semantics

Sprite: Constructor of the Sprite Class, this creates an object of type Sprite given an input size, the sprite sheet to be used.

6 MIS of SpriteSheet Module

6.1 Exported Access Programs

Name	In	Out	Exceptions
SpriteSheet	int, String	-	-

6.2 Interface Semantics

6.2.1 State Variables

SIZE : int
pixels : array of int

6.2.2 Environment Variables

path : String which is used to tell the system the path to the sprite sheet being used

6.2.3 Assumption

6.2.4 Access Program Semantics

SpriteSheet: Constructor of the SpriteSheet class, this sets the path to the desired sprite sheet, sets its size and loads it to be used.

7 MIS of Keyboard Module

7.1 Exported Access Programs

Name	In	Out	Exceptions
Keyboard	int	-	-
keyPressed	KeyEvent	-	-
keyReleased	KeyEvent	-	-

7.2 Interface Semantics

7.2.1 State Variables

playerNum : int
keys : array of boolean

7.2.2 Environment Variables

VK_(UP/DOWN/LEFT/RIGHT/ENTER) : KeyEvent for player 1 controls
VK_(W/S/A/D/G) : KeyEvent used for player 2 controls

7.2.3 Assumption

7.2.4 Access Program Semantics

Keyboard: takes in a parameter for the player number (integer) and is used to initiate a keyboard for that player.

keyPressed: function to check if a key is pressed down while setting the boolean for the specified key that was pressed to true.

keyReleased: function to check if a key is released after having been pressed while also setting the boolean for the specified key that was released to false.

8 MIS of Player Module

8.1 Exported Access Programs

Name	In	Out	Exceptions
Player	int, int, Keyboard	-	-
render	Screen	-	-

8.2 Interface Semantics

8.2.1 State Variables

x : int
y : int
bombBag : int
speed : int

8.2.2 Environment Variables

8.2.3 Assumption

8.2.4 Access Program Semantics

Player: is a function which sets the initial size of the player and chooses the initial sprite to display. Furthermore, it takes an input of a keyboard which will be used to set key controls for that player.

render: takes in a parameter screen which then uses to display the player on to the specified screen.

9 MIS of CollisionTest Module

9.1 Exported Access Programs

Note: the CollisionTest is not yet complete, thus the exported access programs table cannot be filled out.

Name	In	Out	Exceptions
-	-	-	-

9.2 Interface Semantics

9.2.1 State Variables

pixels : array of int
x : int
y : int

9.2.2 Environment Variables

9.2.3 Assumption

9.2.4 Access Program Semantics