Group 7 - Asteroids

1

1.1

Why and How

We used the composite design pattern for our menus. Menu have an hierarchical structure: main-menu, submenu. The composite design pattern makes it easy to recycle code for all the nodes in this structure.

We made an abstract MenuComponent class that is extended by the Menu class. The MenuComponent has a parent MenuComponent attribute and a ArrayList of MenuComponent children. This creates the tree like structure mentioned below.

We used the observer design pattern for the menu-input. One class handles all the input and publishes this to classes that need this information.

We made part of the MenuComponent class that listens to mouse movement and clicks. These changes are then passed to the classes that extend the MenuComponent, the children.

composite

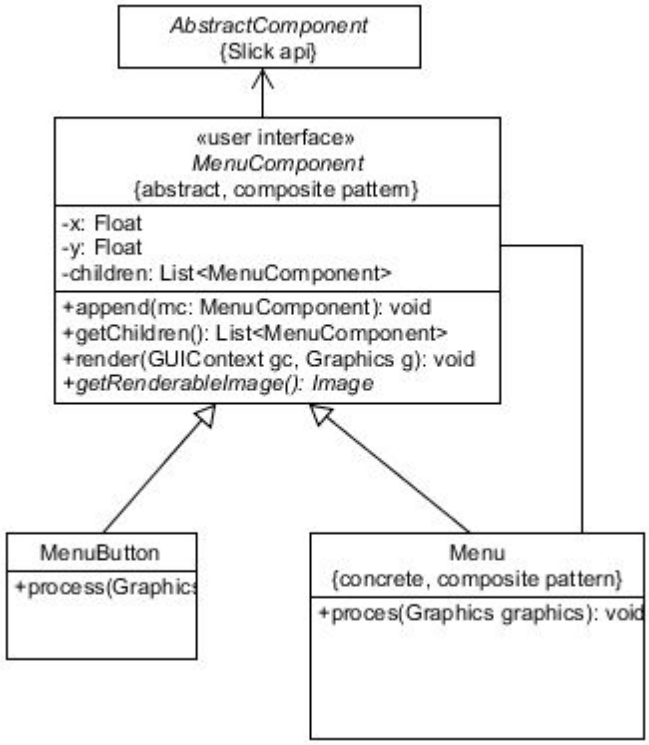
The Composite Pattern allows you to compose objects into tree structures to represent part-whole hierarchies. Composite lets clients treat individual objects and compositions of objects uniformly. - Design Patterns 2 Lecture Slides

observer

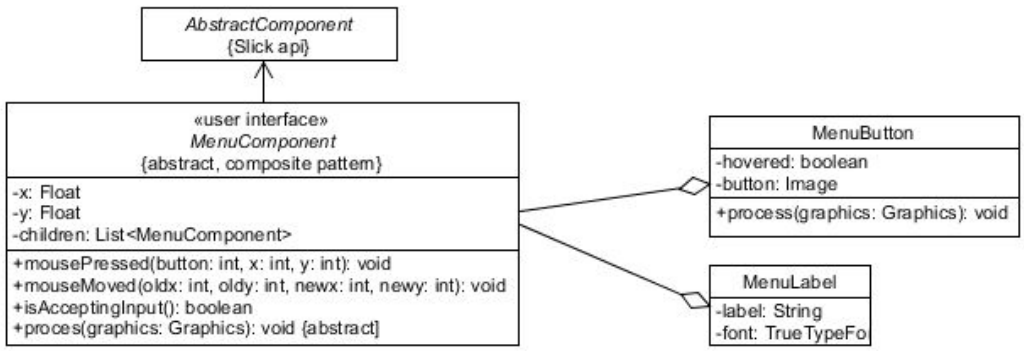
The Observer Pattern defines a one-to-many dependency between objects so that when one object changes state, all of its dependents are notified and updated automatically. - Design Patterns 1 Lecture Slides

1.2 - class diagrams

composite



observer



1.3
sequence diagrams

2
2.1
menu-feature functional requirements
Must Haves:

- The game shall show a menu when it starts (from now on referred to as the start-menu)
- The start-menu shall display three options: singleplayer, multiplayer, exit

- The game shall start with one player if the single player option is selected
- The game shall start with two players if the multiplayer option is selected
- The game shall exit when the exit option is selected
- The menu will be navigated by clicking on the options
- The game shall display menu options as text
- The game shall return to this menu when either one player dies in single player or both players die in multiplayer

Should Haves:

- The menus should have a background OR be an overlay of the game itself

Could Haves:

- The menus can also be navigated with the keyboard

Would Haves:

- The game shall give the menu their own background music
- The game shall give the menu sound effects when navigating and selecting options
- The game shall display animations when an option is selected

menu-feature non-functional requirements

- The menu-feature will be implemented using the slick2d library

2.2

MenuLabel

MenuComponent

-

Process graphics	Graphics element
Have a font	Font element
Have text	Text element

MenuButton

MenuComponent

-

Height and Width	Getters
Have an Image	Image element
Process graphics	Graphics element

MenuComponent

AbstractComponent

Menu

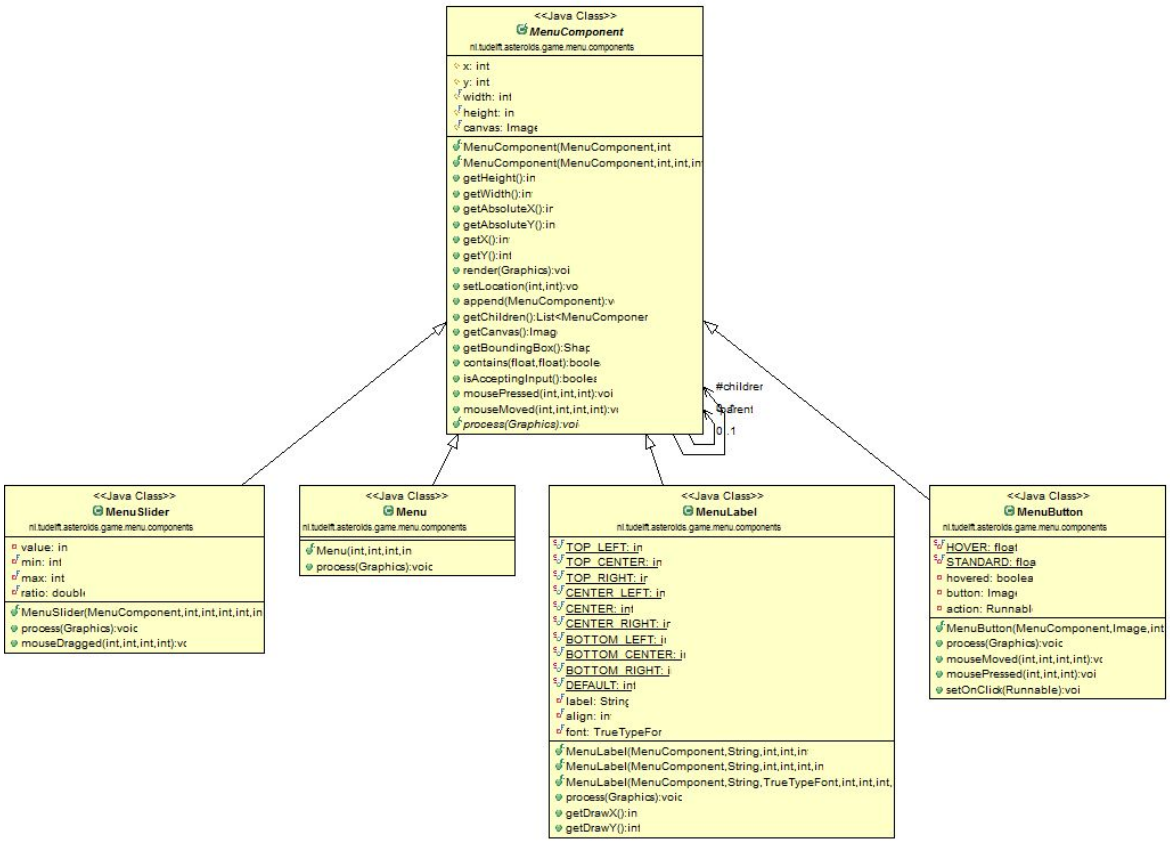
List of children	ArrayList element
Height and Width	Getters
Position	Getters
Render menu on screen	Render method
Handles mouse events	Event methods

Menu

MenuComponent

-

Process graphics	Graphics element



3

3.1

options-feature functional requirements:

Must Haves:

- The options-menu shall be accessed via a new ‘options’ option in the game's start menu
- The game shall take the user to the options-menu (a submenu of the start-menu) when the ‘options’ option is selected
- The options-menu shall display: key-bindings, sound controls, return
- The game shall return to the start-menu when the ‘return’ option is selected
- The game shall start with the selected sound, when the player returns to the start menu and selects the play option
- The game shall start with the selected key bindings, when the player returns to the start menu and selects the play option
- The game shall ask for alternate key bindings when multiplayer is selected. These alternate key bindings will also be available to use for single player when the multiplayer option is not selected.
- The menu will be navigated using the mouse
- The menu will have a background OR be an overlay of the game itself
- The game shall display the options as text OR as buttons with text

Should Haves:

- The game shall save the options selected in the options menu

Could Haves:

- The menu can also be navigated with the keyboard

Would Haves:

- The game shall give the menus their own background music
- The game shall give the menus sound effects when navigating and selecting options
- The game shall display animations when an option is selected
- The options-menu shall display a background selector
- The game shall start with the selected background, when the player returns to the start menu and selects the play option

options-feature non-functional requirements

- The menu-feature will be implemented using the slick2d library

3.2

The CRC cards and UML are the same as for the previous exercise, because the two features were implemented using the same classes.