Description and Time log (from documentation)

5.2 Iteration 2

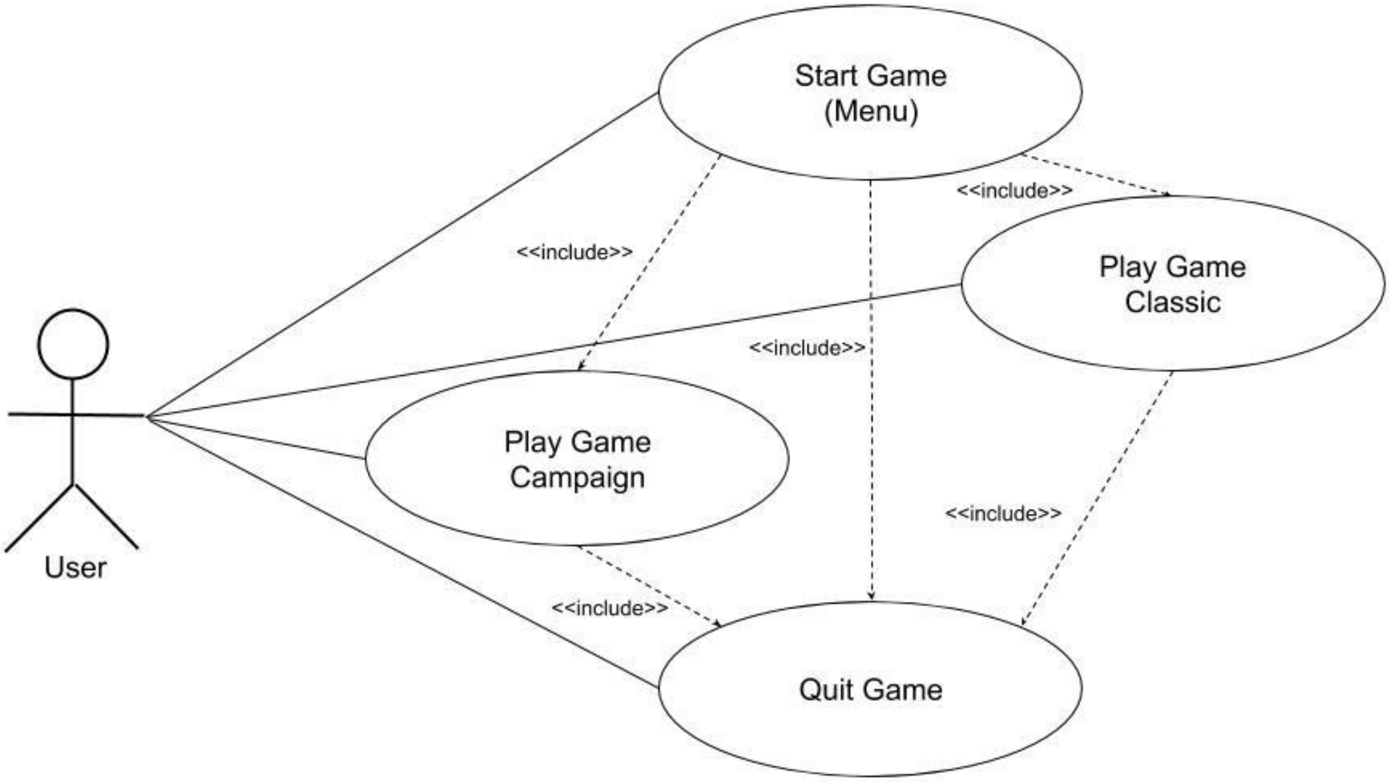
The first implementation of the basic game mode (Vanilla) was done. Before implementing, the Use Case Model and the State Machine Diagram were made using UML. After the implementation, a Class Diagram was also made.

The focus on this Iteration was representing the program using different diagrams and models. All the diagrams are released with this iteration.

7. Time log

Task	Date	Time Estimated	Actual Time
Plan 2nd Assignment	19.02.2019	1:00 H	0:30 H
Use Case Model	20.02.2019	1:30 H	2:30 H
State Machine(basic)	20.02.2019	1:00 H	2:00 H
Implement Vanilla	20.02.2019	3:00 H	1:30 H
Class Diagram	20.02.2019	1:00 H	0:30 H
State Machine(extra)	21.02.2019	1:30 H	2:00 H

Use Case Diagram



Use Case Model

(just the extended version, it didn't make sense to "dumb" it down after I made it)

UC 1 Start Game

Precondition: none.

Postcondition: the game menu is shown.

Main scenario

1. Starts when the user wants to begin a session of the hangman game.
2. The system asks for the user details.
3. The user inputs the details.
4. The system logs the user in and presents the main menu with a title, the option to select mode, view the leaderboard or quit the game.
5. The user makes the choice to select the mode.
6. The system presents the game modes available (Campaign & Free Play).
7. The user chooses the Campaign game mode.
8. The system starts the game in Campaign mode (Execute Use Case 2.1).

Alternative scenarios

3.1 User inputs wrong details

- The system shows an error message and returns to step 2.

5.1 The user makes the choice to quit the game.

- Execute Use Case 3

5.2 The user chooses to view the Leaderboard.

- System shows the Leaderboard and gives the option to return to menu or quit.

7.1 The Campaign game mode is completed

- System shows a message and an option to return to menu or quit.

7.2 The user chooses the Free Play game mode

- Execute Use Case 2.2

UC 2.1 Play Game - Campaign

Precondition: The campaign game mode is selected.

Postcondition: The game is finished, waiting to be shut down.

Main scenario

1. Starts when the user chooses this game mode.
2. The system presents the difficulties available (Peaceful & Hardcore)
3. The user selects the Peaceful difficulty.
4. The system presents the available categories.
5. The user selects a category.
6. The system presents a random word from the category.
7. The user guesses a letter from the word

Repeat step 7 until all the letters are guessed or user runs out of lives.

8. The system shows that the word was guessed.
9. The system asks if the user wants to continue, go back to menu or quit.
10. The user selects "quit the game".
11. The system quits the game (Execute Use case 3).

Alternative scenarios

*. At any point of time, user wants to exit

1. System asks for confirmation

A. 2. User confirms

3. System Exits

B. 2. User doesn't confirm

3. System returns to previous state

4.1 There are no categories left to select from

1. The system shows an error message
2. Go to step 2.

7.1 The user guesses a letter that's not in the word

- The user loses one "life".

7.2 The user guesses a letter that was guessed before

- The system shows a message.

8.1 The user runs out of "lives"

A. If the difficulty is "Peaceful", the word goes back in the list.

B. If the difficulty is "Hardcore", the word doesn't go back in the list.

10.1 The user wants to continue when there are words left in the category.

- Go to step 6.

10.2 The user wants to continue when there are no words left in the category.

1. The system displays a message
2. Go to step 2.

10.3 The user selects "Back to Menu"

- Execute Use Case 1

UC 2.2 Play Game – Free Play

Precondition: The Free Play game mode is selected.

Postcondition: The game is finished, waiting to be shut down.

Main scenario

1. Starts when the user chooses this game mode.
2. The system presents the difficulties available (Classic & Timed Man).
3. The user selects the Classic difficulty.
4. The system presents a random word.
5. The user guesses a letter from the word

Repeat step 5 until all the letters are guessed.

6. The system shows that the word was guessed.
7. The system asks if the user wants to play again, go back to menu or quit.
8. The user selects to quit the game.
9. The system quits the game (Execute Use Case 3).

Alternative scenarios

*. At any point of time, user wants to exit

1. System asks for confirmation

A. 2. User confirms

3. System Exits

B. 2. User doesn't confirm

3. System returns to previous state

3.1 The user selects the Timed Man difficulty

- Same behavior, but in addition, every 10 seconds, one "life" point is lost

5.1 The user guesses a letter that's not in the word

- The user loses one "life".

5.2 The user guesses a letter that was guessed before

- The system shows a message.

6.1 The user runs out of "lives"

- The system shows a "Game over" message.

7.1 The user selects "Back to Menu"

- Execute Use Case 1

7.2 The user selects "Quit"

- Execute Use Case 3

UC 3 Quit Game

Precondition: The game is running.

Postcondition: The game is terminated.

Main scenario

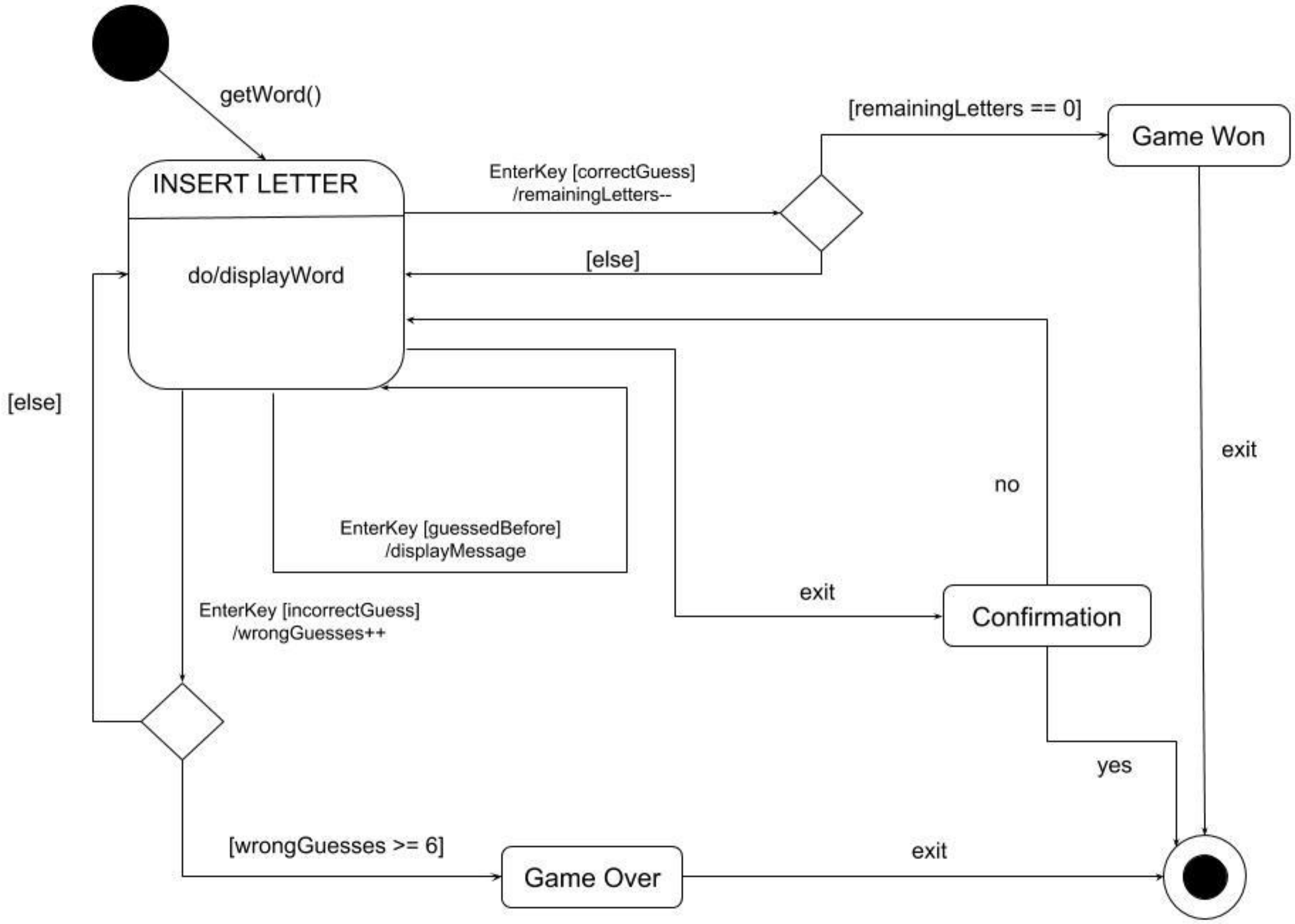
1. Starts when the user wants to quit the game.
2. The system prompts for confirmation.
3. The user confirms.
4. The system shows a message and terminates.

Alternative scenarios

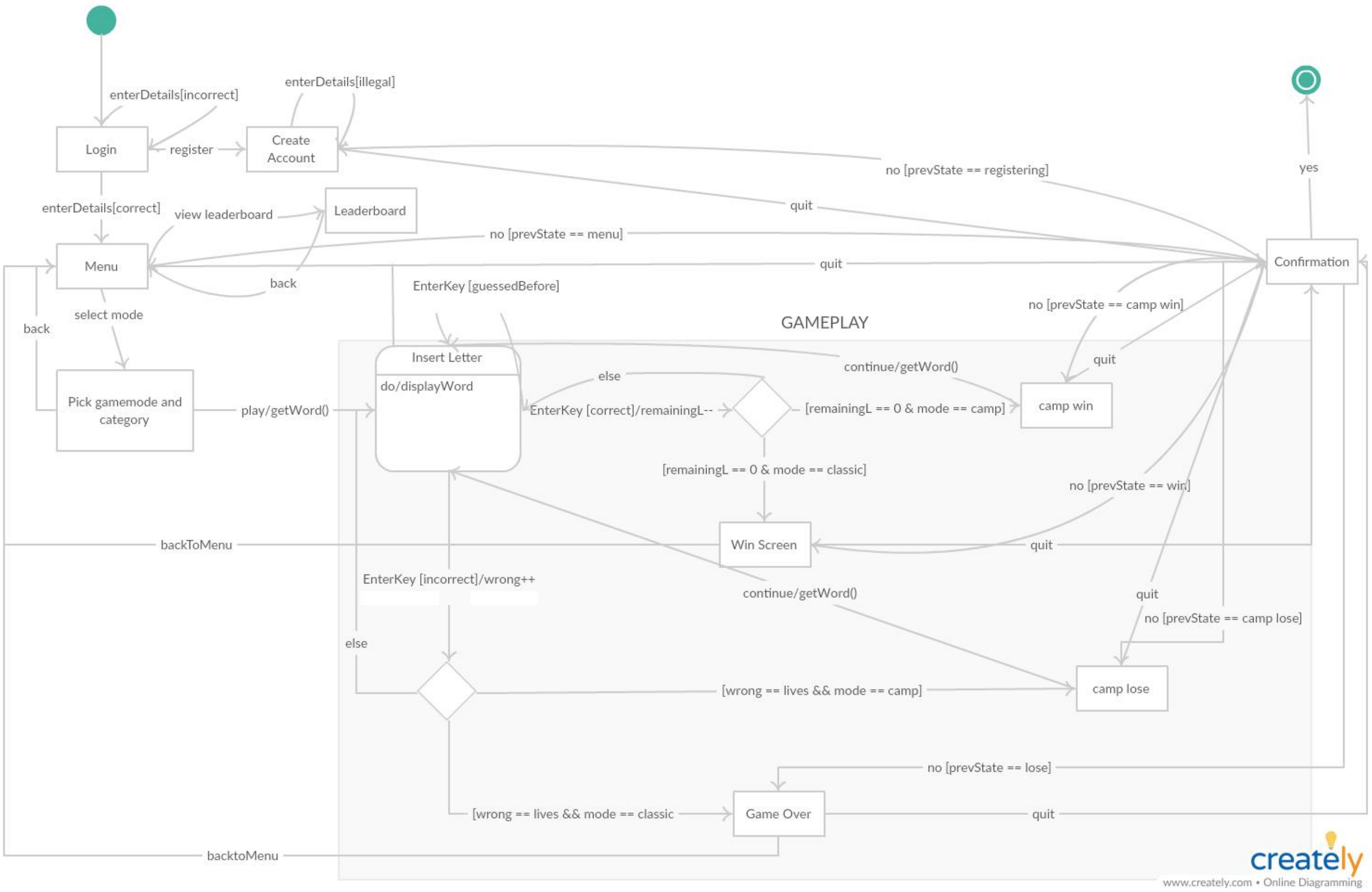
3.1. The user does not confirm

- The system returns to its previous state

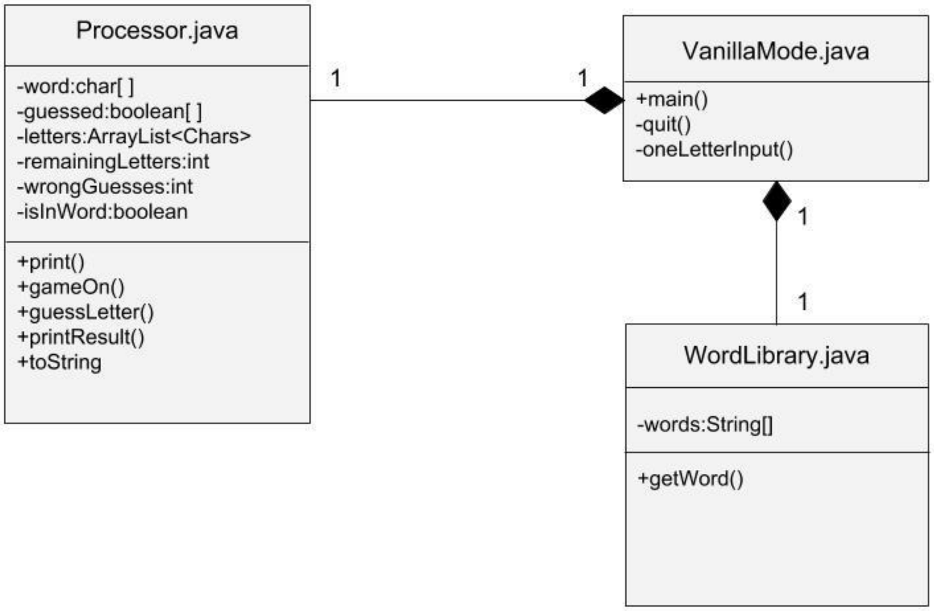
Basic State Machine Diagram



Extended State Machine Diagram



Class Diagram from implementation


Processor.java
-word:char[ ]
-guessed:boolean[ ]
-letters:ArrayList<Chars>
-remainingLetters:int
-wrongGuesses:int
-isInWord:boolean
+print()
+gameOn()
+guessLetter()
+printResult()
+toString
1
1
VanillaMode.java
+main()
-quit()
-oneLetterInput()
1
1
WordLibrary.java
-words:String[]
+getWord()