Helicopter Game

Software Development Project Report

Supervisor
Mohd. Sultan Ahammad

Submitted by
Ketharin Sukhi Tazo(CE18059)
&
Plabon Dibra(CE18060)

Department of Computer Science and Engineering,
Mawlana Bhashani Science and Technology University, Santosh, Tangail-1902

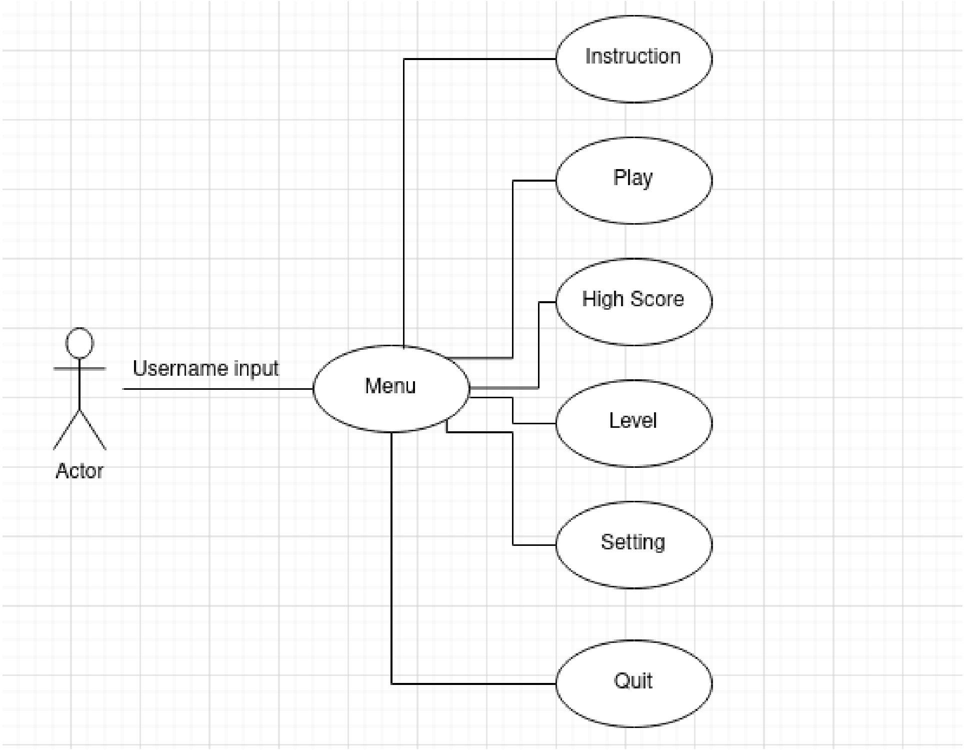
TABLE OF CONTENT

1. Analysis and Domain Modeling

a. Conceptual Model

This is an offline based game. The player who will play will achieve more score he plays the game. Player's final score will make him to take challenge of himself and play again to break the record of the final score.

Here is the use case diagram of this game:



b. System Operation Contracts

In this application when it is clicked it will turn on and open a welcome window
of the game. After that the user have to put his or her name, after putting the name a game display will come up with game menu. In the game menu there will be many options like instructions, play, high score and level with easy medium and hard, there will be other options like sound on or off settings as well if the player don't want to play he or she can quit the game.

Now the game instructions, here the player will know how to control the helicopter. User can use keyboard arrow up and down to control the helicopter.

Up will increase helicopter height and down will decrease helicopter height.

When the user will play, the user can choose level before playing, after choosing the level user will start the game.
While playing user score will be recorded automatically and will be showed later after user finishes the game. Game score of all the previous game played will be recorded so that the user can see the achievement user high scored.

c. Data Model and Persistent Data Storage

Yes, the system need to save data by using text file to store high score for previous game play.

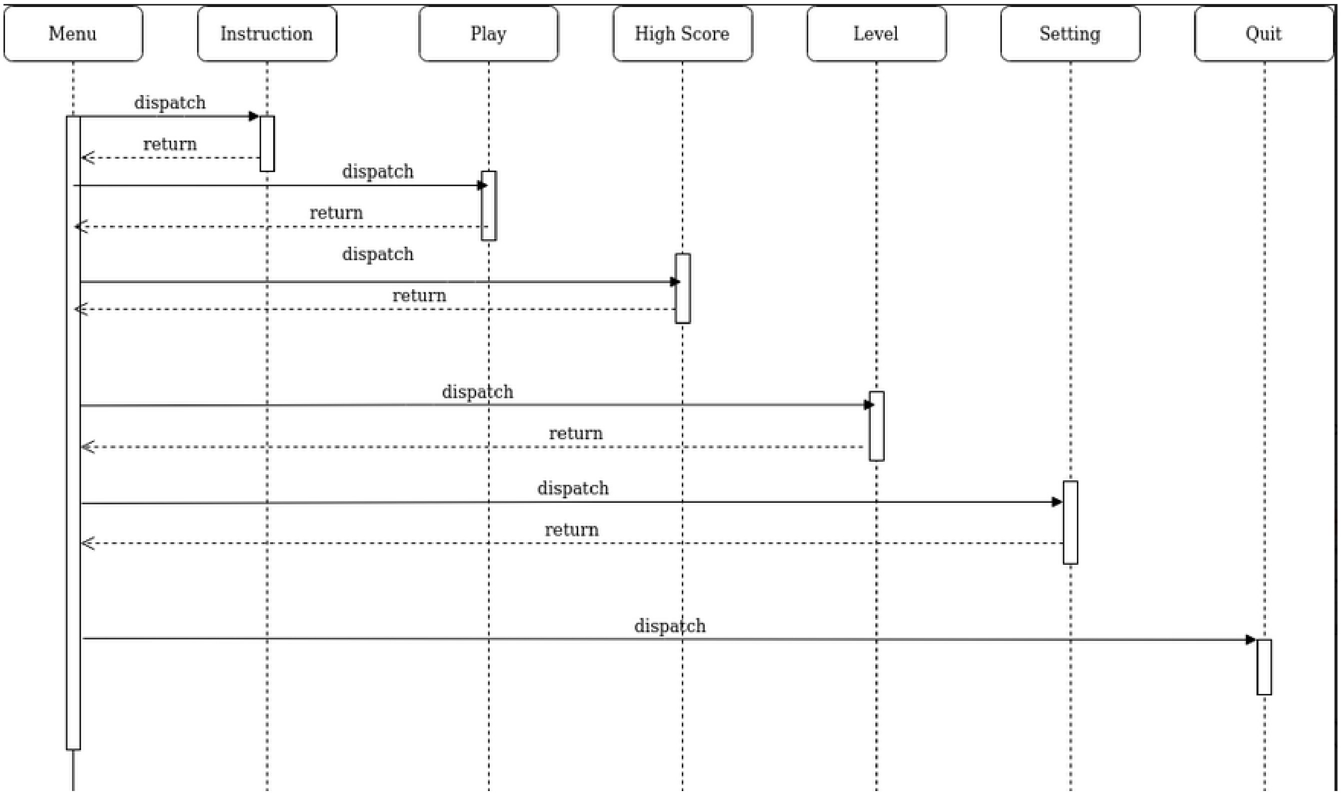
d. Mathematical Model

Not applicable.

e. Project Management

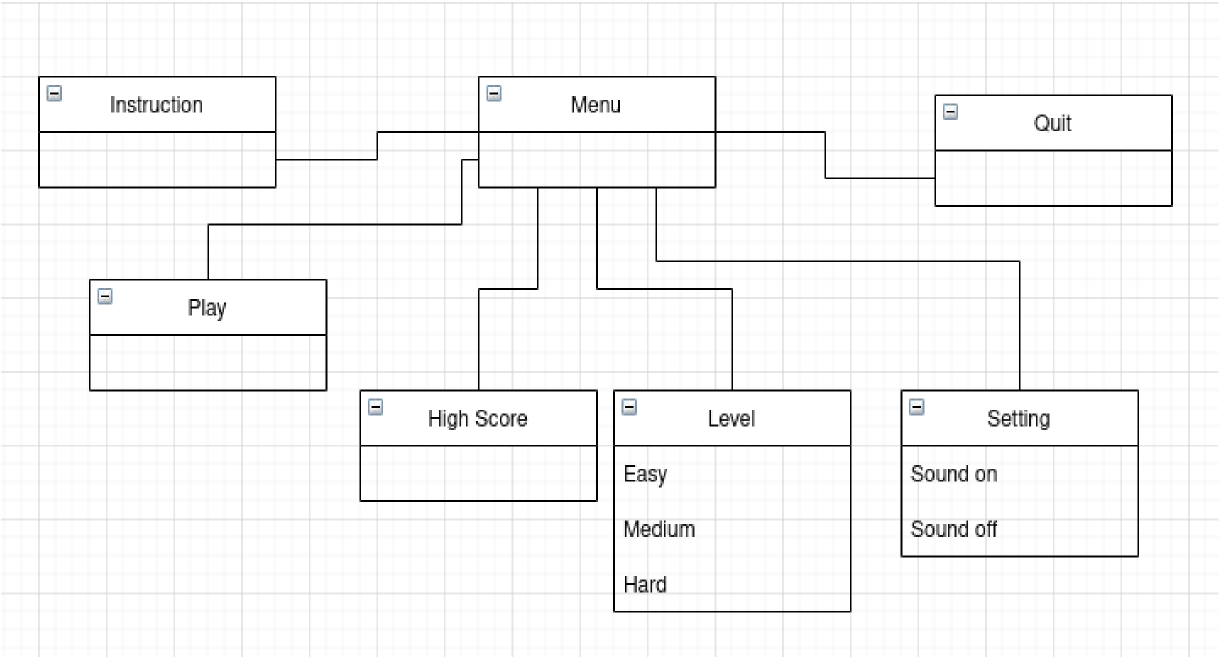
This project is totally controlled by the admin. The admin has the power to update, control the game or even can change the policy of the game.

2. Interaction Diagrams



3. Class Diagram and Interface Specification

a. Class Diagram



b. Data Types and Operation Signatures

In this class diagram menu is main class. There is some sub-class of this menu class. Like instructions, play, high score, level, settings and quit. Level has some object types those are easy, medium and hard. There will be other object types like sound on or off setting as well.

c. Traceability Matrix

Not applicable.

4. Algorithms and Data Structures

a. Algorithms

Algorithm for this application:

```
Step-1:     start
Step-2:     initialize sdl2
Step-3:     variable declaration
Step-4:     reading high score from text
Step-5:     getting images
Step-6:     getting name from user
Step-7:     display welcome image
Step-8:     main menu
Step-9:     if quit then go to step 15
Step-10:    sub menu
Step-11:    creating obstacles
Step-12:    play the game
Step-13:    displaying score
Step-14:    go to step 8
Step-15:    save current high score
Step-16:    end.
```

b. Data Structures

Yes, in this system we use array. We used array to store structural data and collection of primitive data types such as integer.

c. Concurrency

Not applicable.

5. User Interface Design and Implementation

We used SDL2 in our project to implementation. We used this cross-platform development library designed to provide low level access to audio, keyboard, mouse and graphics hardware via in our project. It works with C++.

6. Design of Tests

a. Test Cases

i. Window class
ii. Rect class
iii. Text class
iv. Sound class
v. Game menu
vi. Instruction menu
vii. Play
viii. Level menu
ix. High score menu
x. Setting menu
xi. Quit menu
xii. To integer function
xiii. To string function
xiv. Update score
xv. FPS check
xvi. Moving wall
xvii. Collision
xviii. Save file

b. Coverage

Test coverage means overall test-plan. Test coverage is given details about the level to which the written coding of an application has been tested.

c. Integration Testing strategy

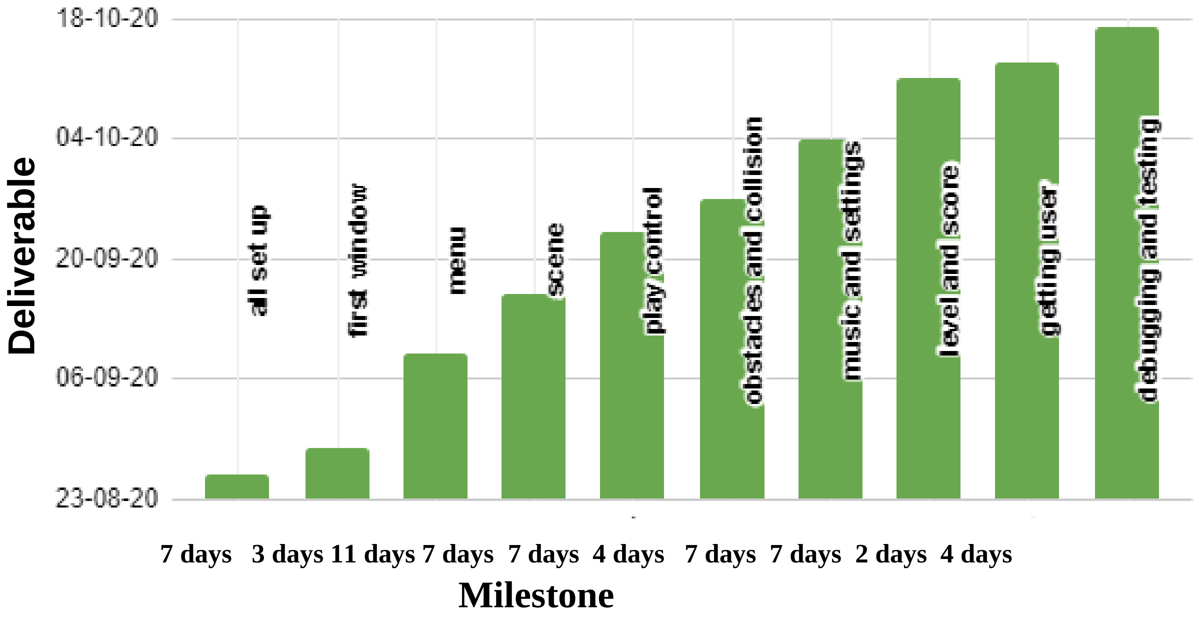
Not applicable.

7. Project Management and Plan of Work

a. Merging the Contributions from Individual Team Members

List	CONTRIBUTIONS BY KETHARIN SUKHI TAZO	CONTRIBUTIONS BY PLABON DIBRA
Requirements Specification	50%	50%
Software design	82%	18%
Coding	7%	93%
Debugging and Testing	7%	93%
Report Preparation	90%	10%

b. Plan of Work



c. Breakdown of Responsibilities

Task	**Date**	**Days**
All set up	26-08-20	1 week
First window	29-08-20	3 days
Menu	09-09-20	11 days
Scene	16-09-20	1 week
Play control	23-09-20	1 week
Obstacles and collision	27-09-20	4 days
Music and settings	04-10-20	1 week
Level and score	11-10-20	1 week
Getting user	13-10-20	2 days
Debugging and testing	17-10-20	4 days

8. References

a. https://www.libsdl.org/download-2.0.php

b. https://sourceforge.net/projects/codeblocks/files/Binaries/20.03/Windows/codeblocks-20.03mingw-setup.exe

19.10.2021

-----------------------------------   -------------------------

Signature of supervisor   Date