Software Requirements Specification

For

Ping Pong Game

14 October 2021

Prepared by

Specialization	SAP ID	Name
CSE-BAO	500075219	Shivani Saran
CSE-BAO	500077646	Utkarsh Sharma
CSE-CSF	500077762	S Sumeet
CSE-CSF	500077853	Udit kumar

Department of Informatics
School Of Computer Science
UNIVERSITY OF PETROLEUM & ENERGY STUDIES,
DEHRADUN- 248007. Uttarakhand

Table of Contents

Revision History

Date	Change	Reason for Changes	Mentor Signature

- # INTRODUCTION

.1 Purpose of the Project

Being one of the earliest computer games, the Pong game is well-known for its simplicity, which makes it suitable for becoming one of the very first problems in Artificial Intelligence and Machine Learning. As we all know, a pong game consists of two paddles on both sides and a ball, and the player has to make sure that he hits the ball with the pad without missing it to win the game and score more.The player can move the paddle up and down. There is a background score as well in the game.

This is a game in which players need to touch the ball with the paddle if the ball will not touch the paddle then the other player will Score points and vice versa.

1. The Main class from which all things are controlled.
2. Ball class that draws the ball and also check its collision with the paddles.
3. Paddle class that draws the paddles and also move the paddles up and down.

This Pong Game Java Project system of the game usually has only one player option. But this game can play by two persons. The game is monotonous as the there is only a single ball that has to be alternated between the opposite sides of the screen without letting the ball miss hitting the pad. The game has intriguing background music whenever we score points. Moreover, it is freeware and can be played by anyone and everyone alike.

Pong Game Java Project Modules

- **Pong module:**– This module complies and runs the game as a single element. This is the main running module.
- **Paddle module:**– This module controls the placement positions of the bricks and the number of bricks.
- **Ball module:**– This module controls the movement of the balls as well as the number of balls.

.2 Target Beneficiary

Playing ping pong helps stimulate mental alertness, concentration and tactical strategy. It is the perfect game for both young and old people for time pass. At the time of pandemic when each and everyone were hanged in their home with no other option to go out they were engaged with their mobile phone to get out of stress and anxiety they faced due to the change in their lifestyle. In order to this many old and new games were played and practiced by them to reduce their level of boredom. Pong game being one of the oldest was adopted by many ,Studies have found that ping pong helps older players improve function of the frontal lobes of the brain, which regulate decision making, problem-solving, and voluntary movements.

1.3 Project Scope

- It Was first Developed by the Atari company in 1972
- Pong is a 2D video game inspired from the sport table tennis
- Features a very basic graphical user interface
- A ball is passed between two paddles controlled by the players
- A player can invoke only two actions
  move the paddle upwards
  move the paddle downwards

Motivation

The Pong game is an excellent test subject because:
It is very simple, it is old, i. e. well-known has been used in Machine Learning and Artificial Intelligence research since the early days of both fields

1.4 References

Kathy Sierra, Bert Bates. (2003) Head first java. O'Reilly Media
SOMASHEKARA, M. T., GURU, D. S., MANJUNATHA. (2017).
Object Oriented Programming with Java. Somashekara media
https://en.wikipedia.org/wiki/Pong

- # PROJECT DESCRIPTION

2.1 SWOT Analysis

Strength:

It is not addictive game, so anyone can play freely anytime.
Best game to spend the time with our friends or family members.
Any age group member can play this game.
Playing ping pong helps stimulate mental alertness, concentration and tactical strategy.

Weakness:
By playing this ping pong game on the screen, people will not go outside for playing table tennis, and this result in loss of physical health.

2.2 Project Features

Pong is a two-dimensional sports game that simulates table tennis.
The player controls an in-game paddle by moving it vertically across the left or right side of the screen.

They can compete against another player controlling a second paddle on the opposing side. Players use the paddles to hit a ball back and forth.

The goal is for each player to reach eleven points before the opponent; points are earned when one fails to return the ball to the other.

2.3 User Classes and Characteristics

classes- pong ,paddle ,puck
paddle have x,y coordinate,width,height,y-axis speed
puck(ball) have x,y coordinate,radius,x and y axis speed,
pong have objects of puck ,paddle.function-keyReleased,keyPressed,draw,setup

2.4 Design and Implementation Constraints

Step 1: Setup() and Draw(), Basic Operation of Processing Code
Step 2: Coordinates in Processing
Step 3: Ball, Object Oriented Programming
Step 4: Setting the Movement of the Ball
Step 5: Colliding the Ball With the Walls
Step 6: Adding the Paddles
Step 7: Moving the Paddles, Key Input
Step 8: Colliding Ball and Paddle

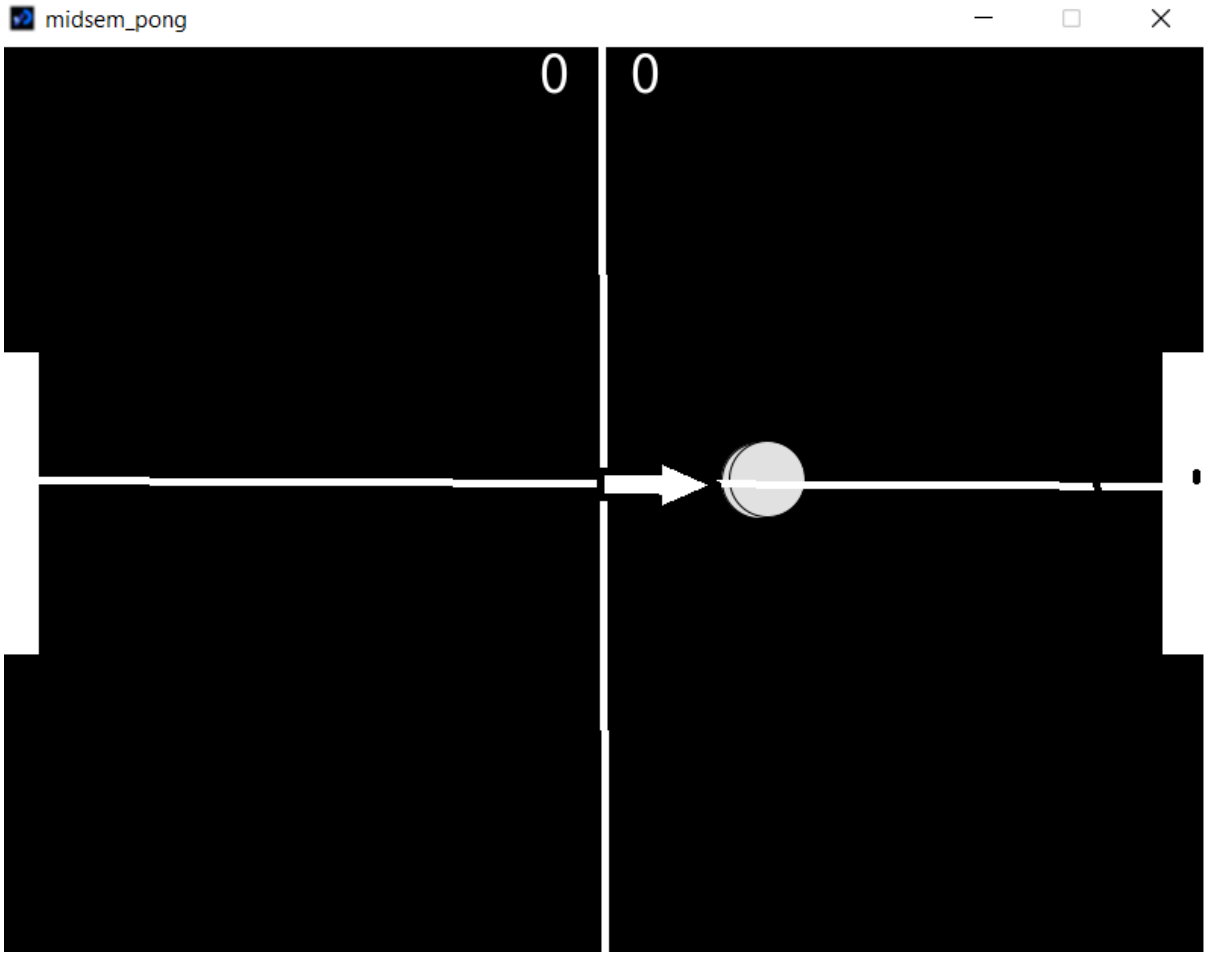
Step 9: Scoring and Starting a New Game
Step 10: Control the Ball Bouncing Direction
2.7 Design Diagrams
2.8 Assumptions and Dependencies

- **SYSTEM REQUIREMENTS**

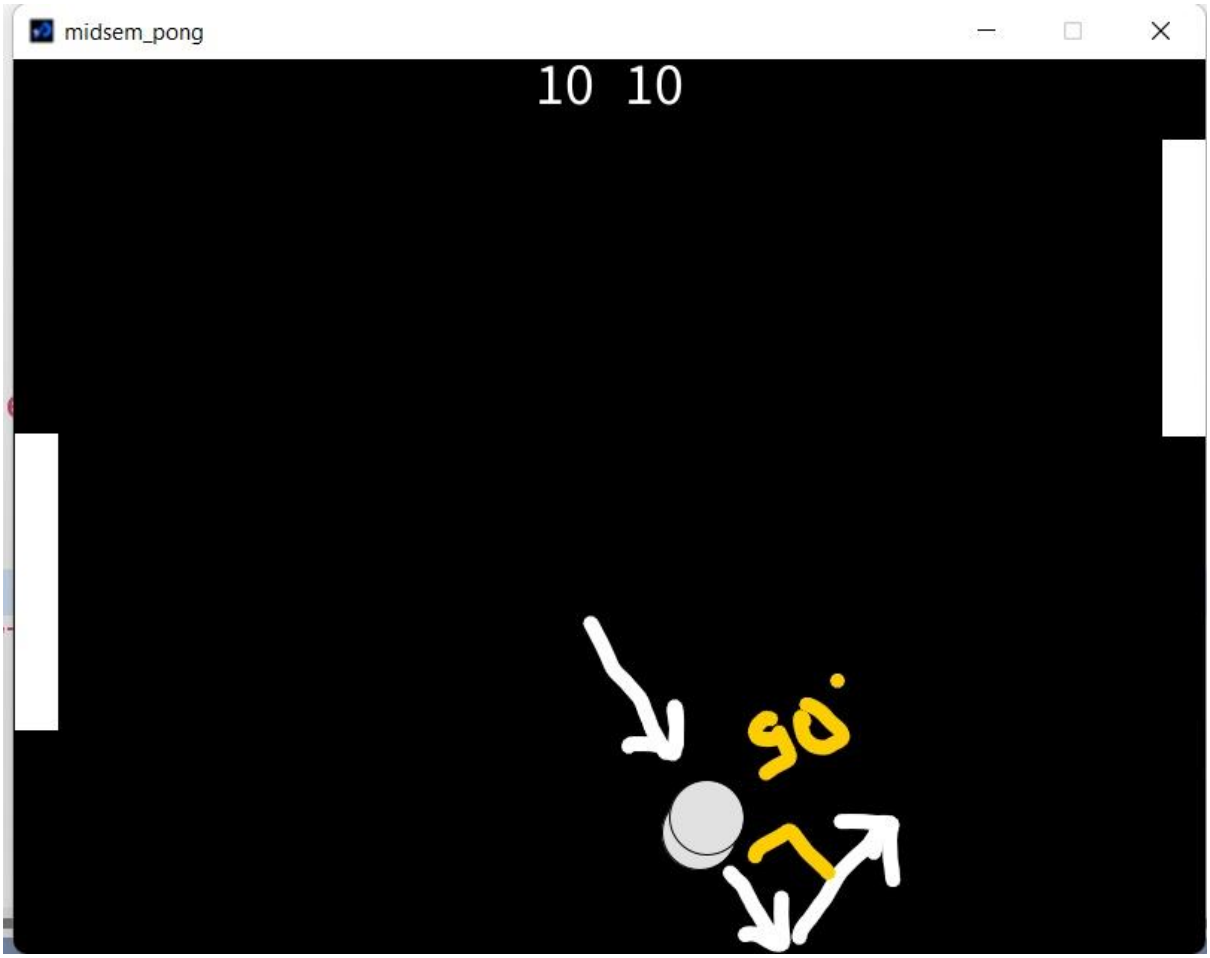
.1 User Interface

The console based application provides a menu allowing for easy control by a keyboard. The menu will allow user to access functionalities such as:

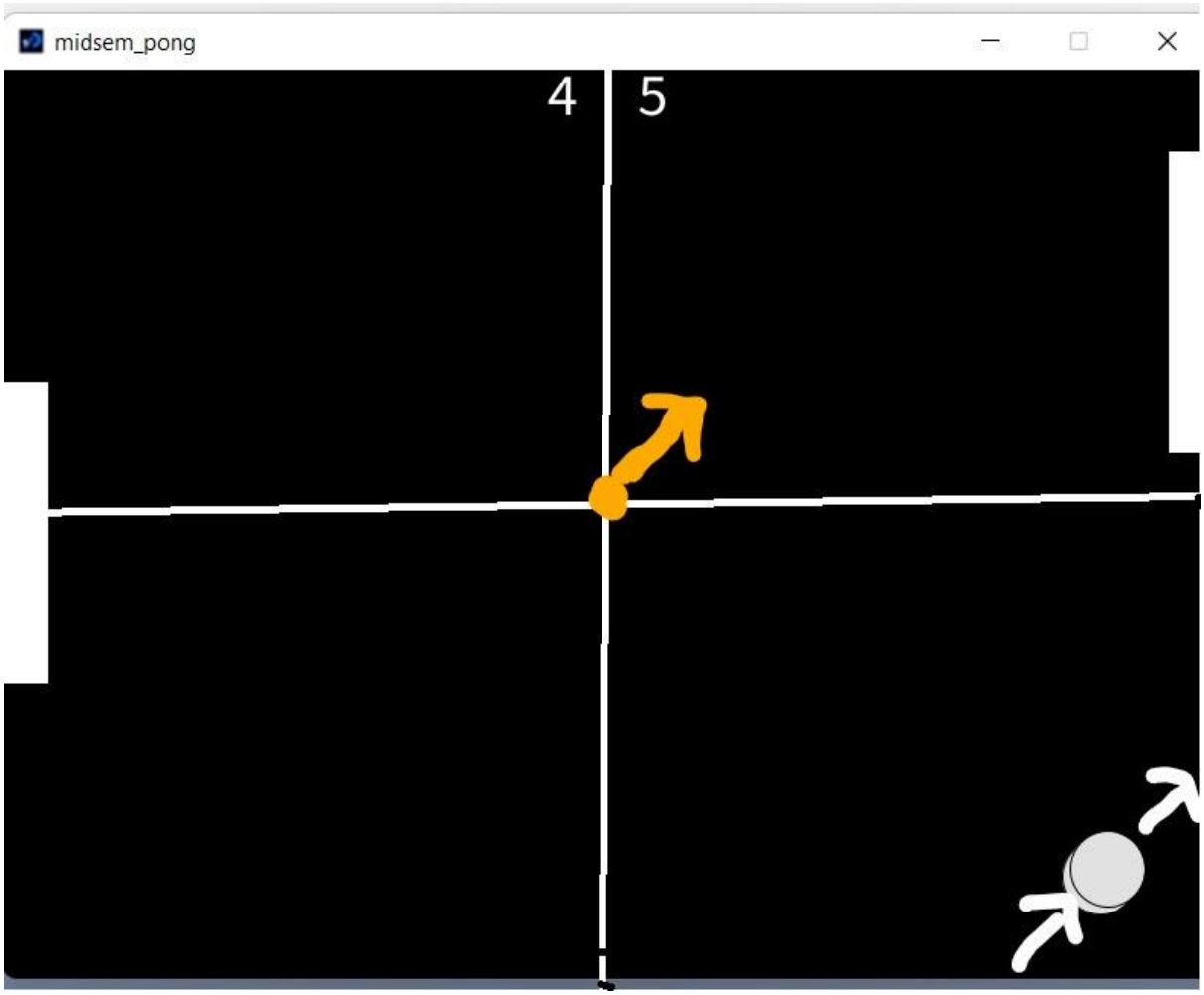
- Make a movement of paddle with the key from keyboard
- Analyze the game
    - When game starts, ball is at center and start moving in right side of the window.



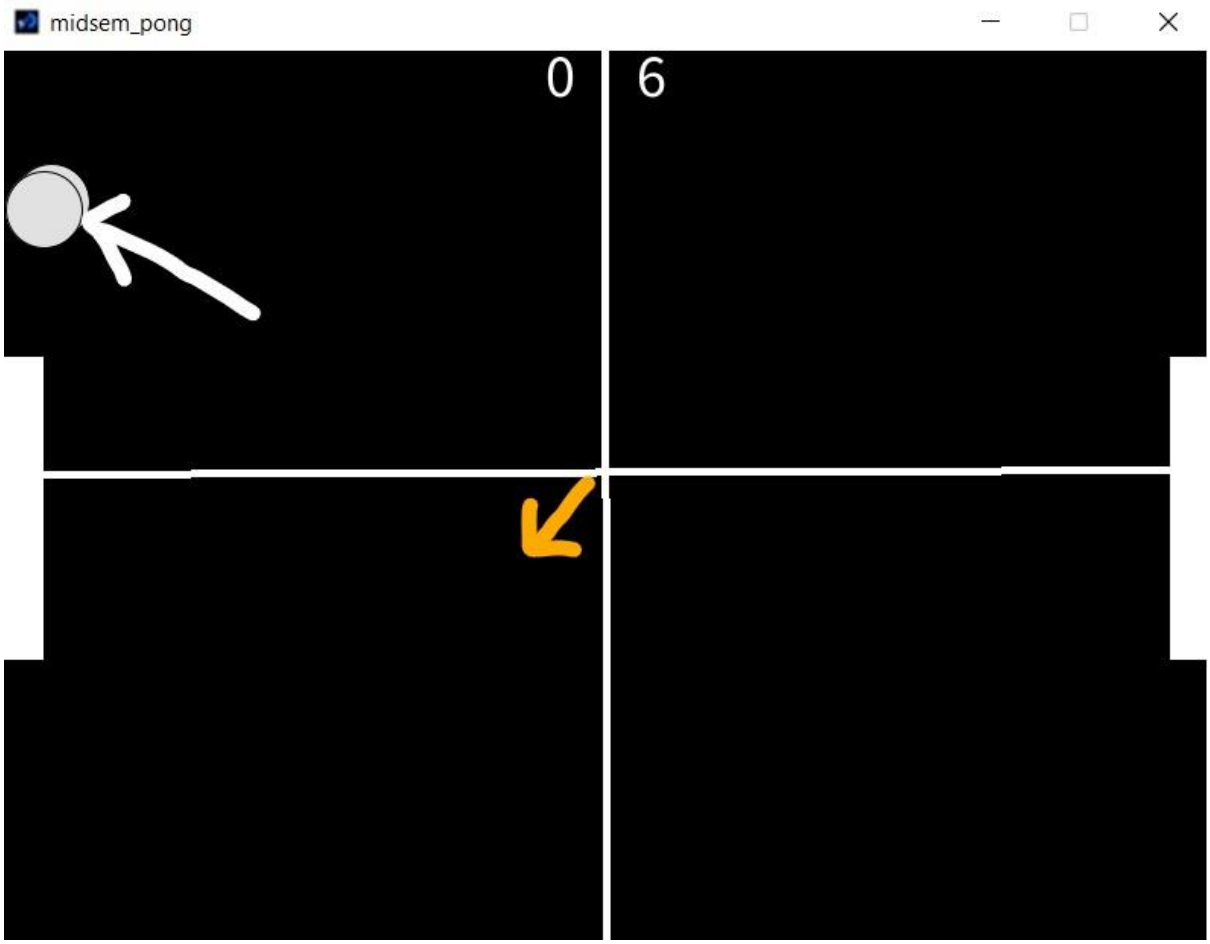
- When ball collide with the top and bottom surface of window, it bounce with 90degree angle.



- When ball goes out of the window form right side (when player could not attend the ball) it starts again from center of the window and move right side with the same angle it passes through the window. And the another player get score.



- When ball goes out of the window form left side (when player could not attend the ball) it starts again from center of the window and move left side with the same angle it passes through the window. And the another player get score.



.2 Software requirement

- Language: C
- Operating system: windows 7 or above
- IDE: processing

4. NON FUNCTIONAL REQUIREMENTS

4.1 Pseudocode

```
//define ball and paddle as global object

    //initialise score to 0 for both player

    function setup(){
       //define size of the window
      //new position of the ball when ball goes out of window
    //speed to ball in x and y direction
      //position of both the paddle

    }

    function draw(){
      //set background colour
    Calling display function of ball class
    Calling move function of ball class

    Calling move function of paddle class

    Calling display function of paddle  class


      if (ball.right() > width) {
        //increase score of left player by 1
        //ball again start with center
      }
      if (ball.left() < 0) {
            //increase score of right player by 1
        //ball again start with center

      }

      if (ball.bottom() > height) {
        //reverse the speed of ball in y direction
      }

      if (ball.top() < 0) {
            //reverse the speed of ball in x direction

      }

      if (paddleLeft.bottom() > height) {
        //stop the left paddle from going outside through bottom of the window while
    moving with      keyboard input.
      }

      if (paddleLeft.top() < 0) {
        //stop the left paddle from going outside through top of the window while moving
    with keyboard input.
      }
```

```
  if (paddleRight.bottom() > height) {
       //stop the right paddle from going outside through bottom of the window while
moving with      keyboard input
  }

  if (paddleRight.top() < 0) {
       //stop the right paddle from going outside through top of the window while
moving with keyboard input.

  }


  if ( ball.left() < paddleLeft.right() && ball.y > paddleLeft.top() && ball.y <
paddleLeft.bottom()){
   / / If the ball gets behind the paddle
  // AND if the ball is in the area of the paddle (between paddle top and bottom)
  // bounce the ball to other direction
  }

  if ( ball.right() > paddleRight.left() && ball.y > paddleRight.top() && ball.y <
paddleRight.bottom()) {
   / / If the ball gets behind the paddle
  // AND if the ball is in the area of the paddle (between paddle top and bottom)
  // bounce the ball to other direction
  }


  textSize is 40
  textAlign at center
  right side player score position(width/2+30,30)
  right side player score position(width/2+30,30)

}

Function keyPressed(){
  if(keyCode == UP){
    //right paddle move in Y direction upwards with some speed
  }
  if(keyCode == DOWN){
       //right paddle move in Y direction downwards with some speed

  }
  if(key == 'a'){
       //left paddle move in Y direction upwards with some speed

  }
  if(key == 'z'){
       //left paddle move in Y direction downwards with some speed

  }
}

Function  keyReleased(){
  if(keyCode == UP){
    //right paddle speed becomes 0 in Y direction.
```

```
  }
  if(keyCode == DOWN){
       //right paddle speed becomes 0 in Y direction.

  }
  if(key == 'a'){
       //left paddle speed becomes 0 in Y direction.

  }
  if(key == 'z'){
      //left paddle speed becomes 0 in Y direction.

  }
}


class Ball {
  float x;
  float y;
  float speedX;
  float speedY;
  float diameter;
  color c;

  Ball(float tempX, float tempY, float tempDiameter) {
    //constructor method
  }

  function move() {
//add speed to loaction
  }

  function display() {
    //set the drawing colour
    //draw a circle
  }


  function left(){
    //return x-diameter/2
  }
  function right(){
    //return x+diameter/2;
  }
  function top(){
    //return y-diameter/2;
  }
  function bottom(){
    //return y+diameter/2;
  }

}
```

```
class Paddle{

  float x;
  float y;
  float w;
  float h;
  float speedY;
  float speedX;
  color c;

  Paddle(float tempX, float tempY, float tempW, float tempH){
//constructor method
  }

  function move(){
    //y += speedY;
    //x += speedX;
  }

  function display(){
  //colour for the paddle
   // rect(x-w/2,y-h/2,w,h);
  }


  float left(){
    //return x-w/2
  }
  float right(){
   // return x+w/2
  }
  float top(){
  //  return y-h/2
  }
  float bottom(){
    //return y+h/2
  }
}
```

4.1 Security Requirements

It is not asking for any personal information so there is no any security issue.

4.2 Software Quality Attributes

The software is in the form of windows executable file and, it will only work on system operating on windows OS. However, the program can be compiled on any other operating system, to get the file in desired file extension.

Availability: The scheduler will tell whether it will execute the tasks it is assigned to perform.

Adaptability: The scheduler is capable of being adapted for different specific environments without applying actions other than those provided for this purpose for the software considered.

Reliability: Measure if the product is reliable enough to sustain in any condition. Should give the correct results consistently. Product reliability is measured in terms of working of the project under different working environments and different conditions.

Maintainability: Different versions of the product should be easy to maintain. For development, it should be easy to add code to the existing system, should be easy to upgrade for new features and new technologies from time to time.

Usability: This can be measured in terms of ease of use. The application should be user-friendly. It should be easy to learn. Navigation should be simple.

Portability: This can be measured in terms of Costing issues related to porting, Technical issues related to porting, and Behavioral issues related to porting.

Correctness: The application should be correct in terms of its functionality, calculations used internally and the navigation should be correct. This means that the application should adhere to functional requirements.

Flexibility: Should be flexible enough to modify. Adaptable to other products with which it needs interaction. Should be easy to interface with other standard 3rd party components.

Interoperability: It is capable enough for easy exchange of data or services with other systems. Different modules can work on this scheduler with different platform, different databases and protocol conditions.

Reusability: Software reuse is a good cost-efficient and time-saving development method. Different code library classes should be generic enough to be easily used in different application modules. Divide the application into different modules so that modules can be reused across the application.

Robustness: The code is able to cope with errors during execution and cope with erroneous input.

Testability: The system should be easy to test and find defects. If required, it should be easy to divide into different modules for testing.

5. OTHER REQUIREMENTS

5.2 Appnedix B: Analysis Model