Final Specification Document

Group Members : Bishesh Tuladhar, Pukar Mahat

Fast and Furious Game

Design Specification

Fast and Furious game is a fun mini game. The user will control the car and the objective of the game is the avoid other cars in the traffic. The speed of the car and traffic will increase as the time of the game increases.

The game also has score and medal system. The user uses A and D key to move the car left and right respectively. There are two AI car which will respawn to a new random position after every time it goes out of bound from the canvas.

For the collision detection of the player car and the AI car, an hit box was created which is rectangle. Its function is to check if two of the hit boxes collide with each other.

Have Fun !!

Fast and Furious Game Use Case

|  |  |  |
| --- | --- | --- |
| Steps | User’s Action | System’s Response |
| 1. | User presses start |  |
| 2. |  | The game starts and the car is moving straight. |
| 3. | User presses A or D in keyboard |  |
| 4. |  | If A is pressed car moves to the left and if D then right. |
| 5. | User keeps going and does not hit other cars. |  |
| 6. |  | Game speeds up as greater distance is covered. |
| 7. | User hits other cars |  |
| 8. |  | The game stops and the distance covered, and the prize based on distance is displayed. Either Gold, Silver, or Bronze. |
| 9. | User Clicks the Start Button |  |
| 10. |  | Game Resets |
| 11. | Game Ends |  |

Pseudo Code

public void Reset()

{

All the value are set to default

All the object are set to default position

Timer is started (game cycle begins)

}

public bool intersectsWith(Rectangle ob1, Rectangle ob2)

{

Checks if the Rectangle ob1 and Rectangle ob2 collide with each other

}

void timer\_Tick(object sender, EventArgs e)

{

Cycle of the game begins

Score++;

RoadTrack animation

Move car left (if the user gives left command and the car is within bound)

Move car right (if the user gives left command and the car is within bound)

Stop car (if no command is issued)

AI car animation and changes its source

}

private void moveCar(object sender, KeyEventArgs e)

{

Move car to left if the ‘A’ key is pressed and the car is within bound

Move car to right if the ‘D’ key is pressed and the car is within bound

}

private void stopCar(object sender, KeyEventArgs e)

{

Stop car if no key is pressed

}

private void changeAI1()

{

Changes the AI1 car to random new source

}

private void changeAI2()

{

Changes the AI2 car to random new source

}

private void gameOver()

{

Timer cycle ends;

Score is calculated

Trophy is displayed according to the score;

}

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

Resets the game when clicked;

}

private void DevelopStats()

{

Displays the stats of the object in the game for the developer;

}