Periode 3

De Gokkers

Groep 6 | Terheijdenseweg 350 Breda

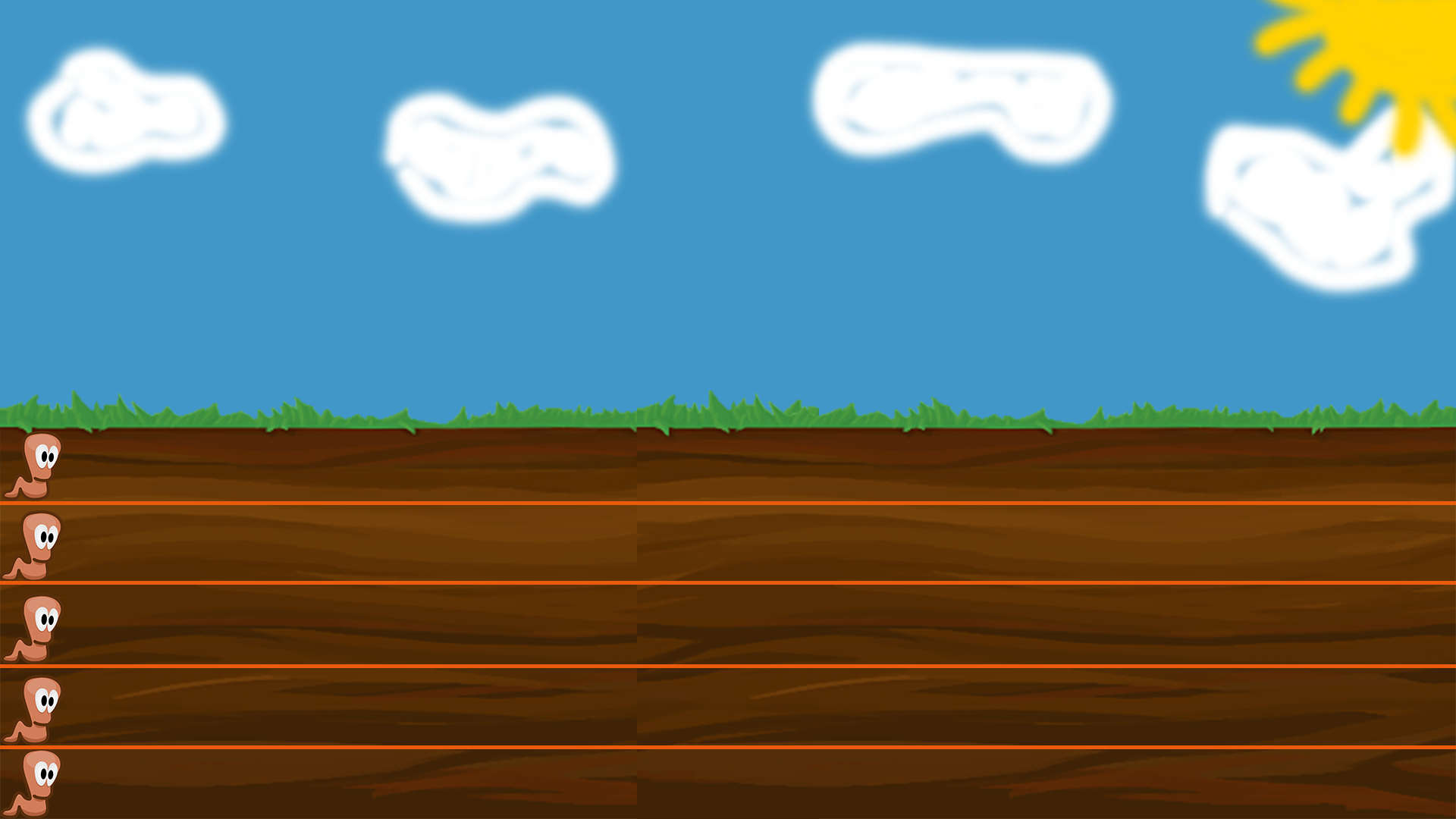
sjabloon

Kevin iwan tomasz GROEP 6 12-02-2017 T/M 10-04-2017

Contents

[Heading 1 2](#_Toc474764288)

# Lay-out van de race baan en interface.



Onze interface zou bestaan uit twee kolommen in eerste kolom kun je je bet inzetten en in tweede kolom zie je de uitslag of je wel of niet hebt gewonnen.

Die zou zo ongeveer zien:



Hier in worden alle gegevens van de player gedisplayd en ook gecheckt of je genoeg geld hebt om in te zetten.

# Classes

We gaan gebruik van drie classes maken die als volgende heten.

# UndergroundWorm

# Player

# Bet

Class Player:

class Player

{

public string Name; // De naam van de gokker

public Bet MyBet; // Een instantie van Bet()

public int Cash; // Het saldo van de gokker

public void UpdateLabels()

{

//Verander mijn label in de omschrijving van mijn weddenschap.

//Verander de label op mijn radioknop zodat deze mijn saldo laat zien.

//(Bijv. “Lidy heeft 43 euro.”)

}

public bool PlaceBet(int amount, int dog)

{

//Plaats een nieuwe weddenschap en sla het op in de variabele MyBet.

//Retourneer een true als de gokker genoeg geld heeft om te wedden.

//Onderstaande regel staat er tijdelijk om foutmeldingen te voorkomen.

//Haal deze regel later weg.

return true;

}

public void ClearBet()

{

//Maak de weddenschap leeg.

}

public void Collect(int Winner)

{

//Betaal mijn weddenschap uit.

//Maak mijn weddenschap leeg.

//Werk mijn labels bij.

}

}

Class UndergroundWorm

class Undergroundworm

{

public int RaceTrackLength; //De lengte van de renbaan

public PictureBox MyPictureBox = null;

public Random Randomizer; //Een instantie van Random (= Willekeurig)

public bool Run()

{

//Ga willekeurig 1, 2, 3 of 4 posities naar voren.

//Werk de positie van PictureBox bij op het formulier.

//Geef de waarde ‘true’ terug als ik de race win.

}

public void TakeStartingPosition()

{

//Wijzig mijn locatie naar de startlijn.

}

}

Class Bet

class Bet

{

public int Amount; //Het bedrag van de weddenschap.

public int Worm; //Het nummer van de hond waarop weddenschap is afgesloten.

public Player Bettor; //De gokkker die gewed heeft.

public string GetDescription()

{

//Retourneer een string die aangeeft wie de weddenschap heeft gedaan,

//voor welk bedrag er is gewed en op welke hond er is gewed.

//Bijv. “Sietse wedt 8 euro op hond 4”.

//Als het bedrag 0 is, is er geen weddenschap geplaatst.

//De string die dan geretourneerd wordt is bijv.

//“Sietse heeft geen weddenschap geplaatst.”)

}

public int PayOut(int Winner)

{

//De parameter van deze methode is de winnaar van de race.

//Als de hond gewonnen heeft, retourneer dan het bedrag dat gewed is

//Anders, retourneer het tegengestelde van het gewedde bedrag.

}

}

# Worm gaat rennen

De klasse UndergroundWorm houdt de positie op de renbaan bij tijdens de race. Het werkt ook steeds de locatie van de PictureBox (die de hond vertegenwoordigt) bij. Elke instantie van UndergroundWorm gebruikt een veld MyPictureBox om te refereren naar de PictureBox controller op het formulier die de afbeelding van de hond toont.

Maak een variabele ‘distance’ aan die de afstand bevat die de hond naar voren moet rennen. Je code zal dan de locatie van MyPictureBox updaten door de variabele distance aan de X-waarde toe te voegen:

Je moet er voor zorgen dat het formulier de juiste afbeelding aan d juiste hond toekent.

MyPictureBox.Location = new Point(

MyPictureBox.Location.X + distance, MyPictureBox.Location.Y);

Logische ontwerp:

