## **ОПИС НА АПЛИКАЦИЈАТА**

Апликацијата претставув single-player игра, во која главната цел е уништување на сите топчиња во дадено ниво, за дадено време. Играта се состои од 5 нивоа, кои стануваат потешки како што играчот прогресира. При изработка на играта се водевме по нејзината оригинална верзија https://www.miniclip.com/games/bubble-trouble/en/

## **УПАТСТВО ЗА УПОТРЕБА**

### **2.1 Почеток на апликацијата(слика1)**

Апликацијата започнува со главно мени, на кое се понудени опциите:

* **Start Game** - со клик на ова копче се започнува нова игра
* **Show Controls** - со клик на ова копче во нова фома се прикажуваат контролите со кои играчот треба да ја игра играта.
* **?** - со клик на ова копче се добиваат повеќе информации за целта на играта
* **Копче за звук** – со клик на ова копче играчот има опција да го исклучи/вклучи звукот на играта.

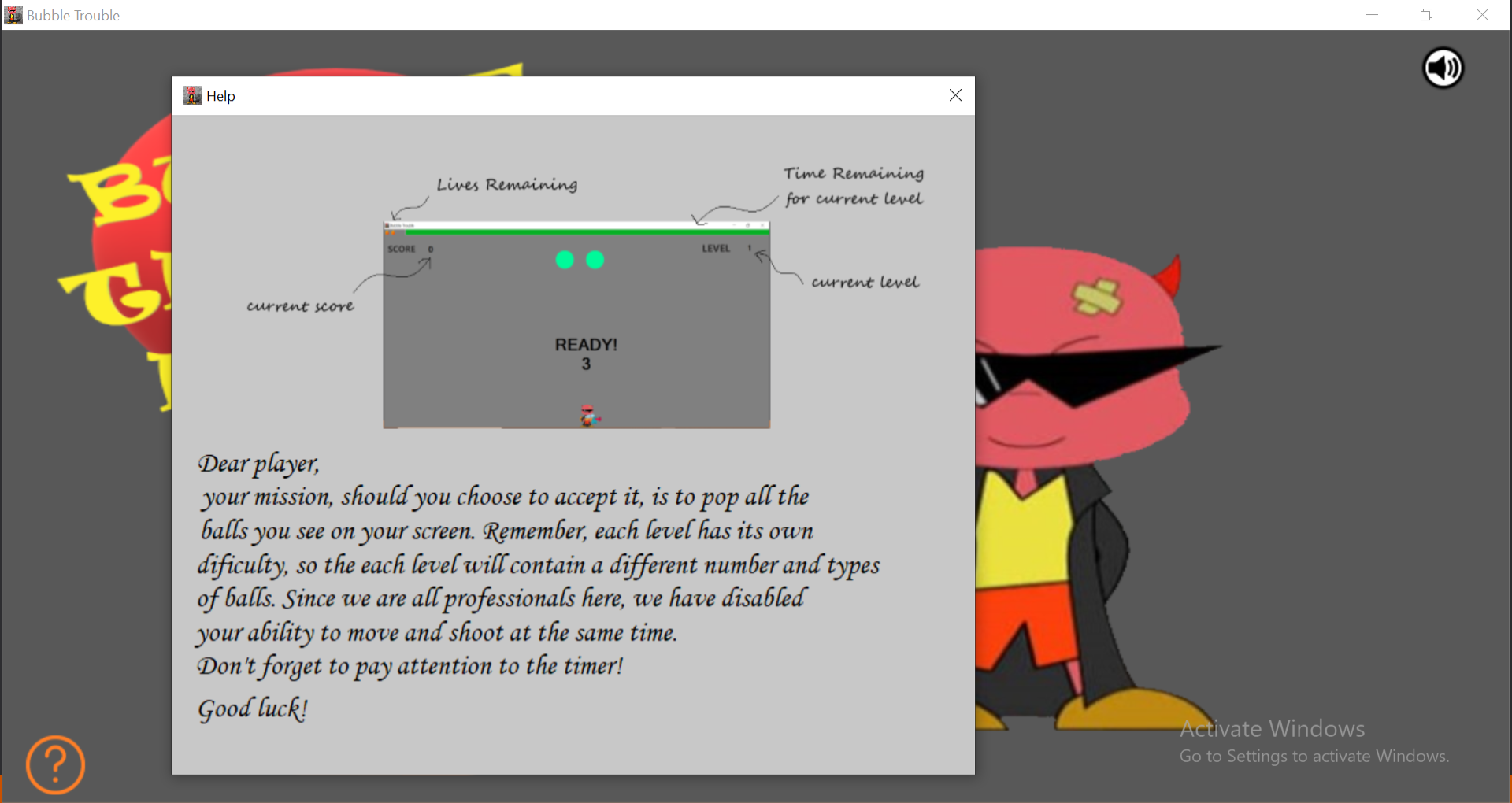
Слика1

### **2.2 Нова игра**

Со клик на копчето Start Game се започнува нова игра, со што се почнува од Level1. Играчот ја започнува играта со 3 животи.

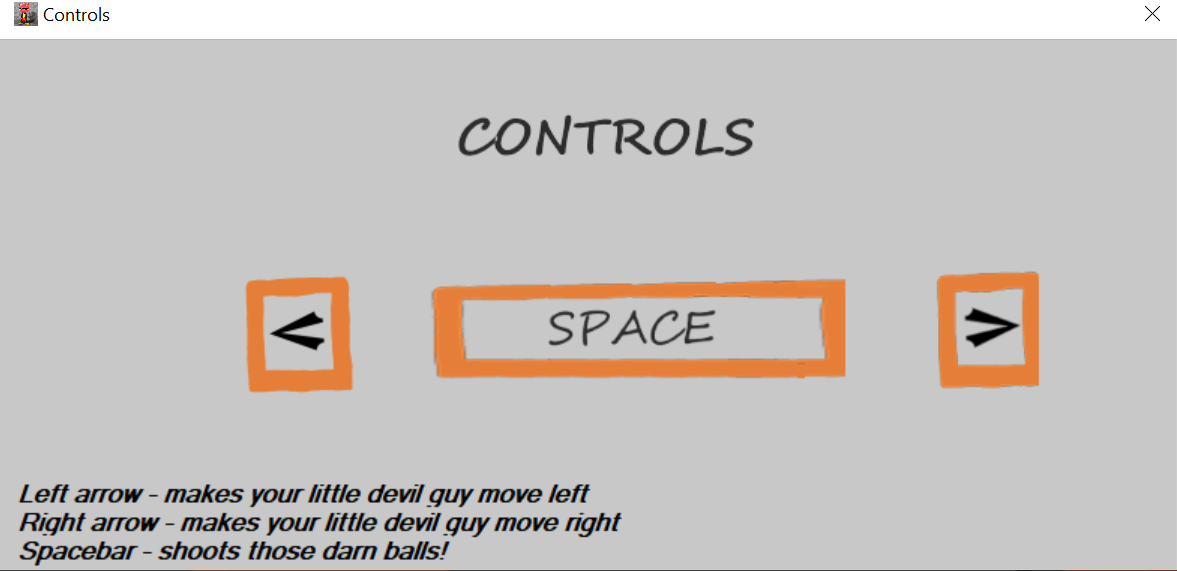
### **2.3 Правила на игра**

* Играчот има задача да ги избегнува топките, ако не успее тогаш губи живот.
* Ако истече тајмерот, играчот губи живот.
* Ако играчот ги изгуби сите животи, играта завршува.
* Играчот стигнува до следното ниво, ако ги уништи сите топки од даденото ниво.

\*овие правила се објаснети со клик на прашалникот во главното мени(слика1)

слика2

### **2.4 Show Controls (слика3)**

Со клик на ова копче се отвара нова форма која ги покажува контролите за играње

слика3

### **2.5 Gameplay**

## **3. ПРЕТСТАВУВАЊЕ НА ПРОБЛЕМОТ**

### **3.1 Потребни структури**

За создавање на ваква 2D игра, потребни се повеќе видови на класи кои ќе чуваат податоци за целата игра, посебните нивоа, играчот, препреките и топките кои се главната карактеристика на играта. Покрај тоа се потребни функции кои ке обезбедат движење на топката и играчот, функција за пукање, функции за судир меѓу топка и играч, и топка и препрека.

### **3.2 Имплементација**

#### **3.2.1 Game**

Оваа е главната класа на оваа апликација, бидејќи таа контактира со главната форма. Оваа класа има одговорност да биде медиум меѓу класата Level и главната форма. Во нејзе ги чува димензиите на екранот и моменталната инстанца на Level.

#### **3.2.2 Level**

Класата Level е апстрактна класа од која наследуваат 5 други класи: LevelOne, LevelTwo, LevelThree, LevelFour и LevelFive. Во оваа апстрактна класа се чуваат следните податоци.

private Point StartingPosition; - (x,y) координатите на почетната позиција на играчот

protected int Width, Height; - големината и ширината на екранот

protected int[] YShootCoordinatesForGivenX; - низа од соодветните Y координати за дадено Х

public List<Ball> Balls; - листа од топки во нивото

public List<Obstacle> Obstacles; - листа на препреки во нивото

protected int TimeLimit; - временски лимит на нивото

public int LevelID { get; set; } - идентификација на нивото (1-5)

Следно се имплементирани неколку функции кои важат за сите класи коишто ќе наследат од класата Level:

public void DrawLevel(Graphics g) - основна функција која овозможува целосно исцртување на левелот. Во нејзе се повикуваат Draw() функциите на сите визуелни елементи односно : играч, препреки и топки.

public void MovePlayer(int dx, int dy) - му овозможува поместување на играчот

public void PlayerShoot() - му овозможува на играчот да пукне

public void MoveBalls() - овозможува придвижување на топките

public void RemoveShotAt() - функција која проверува дали е погодена некоја топка, и соодветно ја отсртанува.

public void AddBall(Ball ToAdd) - метод за додавање топки

public void AddObstacle(Obstacle ToAdd) - метод за додавање препреки

public void PreprocessShootingYs() - функција која ги препроцесира У координатире за дадено Х. Поточно, бидејќи во секое ниво имаме различен број и вид на препреки, кога играчот се наоѓа под некоја препрека и сака да испука треба да се осигураме дека тој истрел нема да помине низ препреката и да удри топка која се наоѓа над истата. Дополнително ова ни е и начин како да знаеме до каде да ја исцртаме линијата која го претставува истрелот

Преостануваат уште две апстрактни методи, кои всушност се користат за давање уникатност на левелите што ќе наследат:

public abstract void GenerateObstacles(); - метод кој генерира препреки

public abstract void GenerateBalls(); - метод кој генерира топки

#### **3.2.3 Player**

Player е класа која содржи информации за моменталните положба и моменталните поени на играчот. Низ текот на целата игра се користи една и единствена инстанца на класата, за што е искористен Singleton design pattern. Во оваа класа се имплементирани функционалностите за движење на играчот, намалување на неговите животи, апдејтирање на поените, проверка за судир со топка, и пукање.

#### **3.2.4 Ball**

Ball е апстрактна класа која се користи да се создадат различните видови топки GreenBall, BlueBall и RedBall. Главните функционалности се имплементирани во апстрактната класа како што се движење на топката и проверка за судир со препрека.