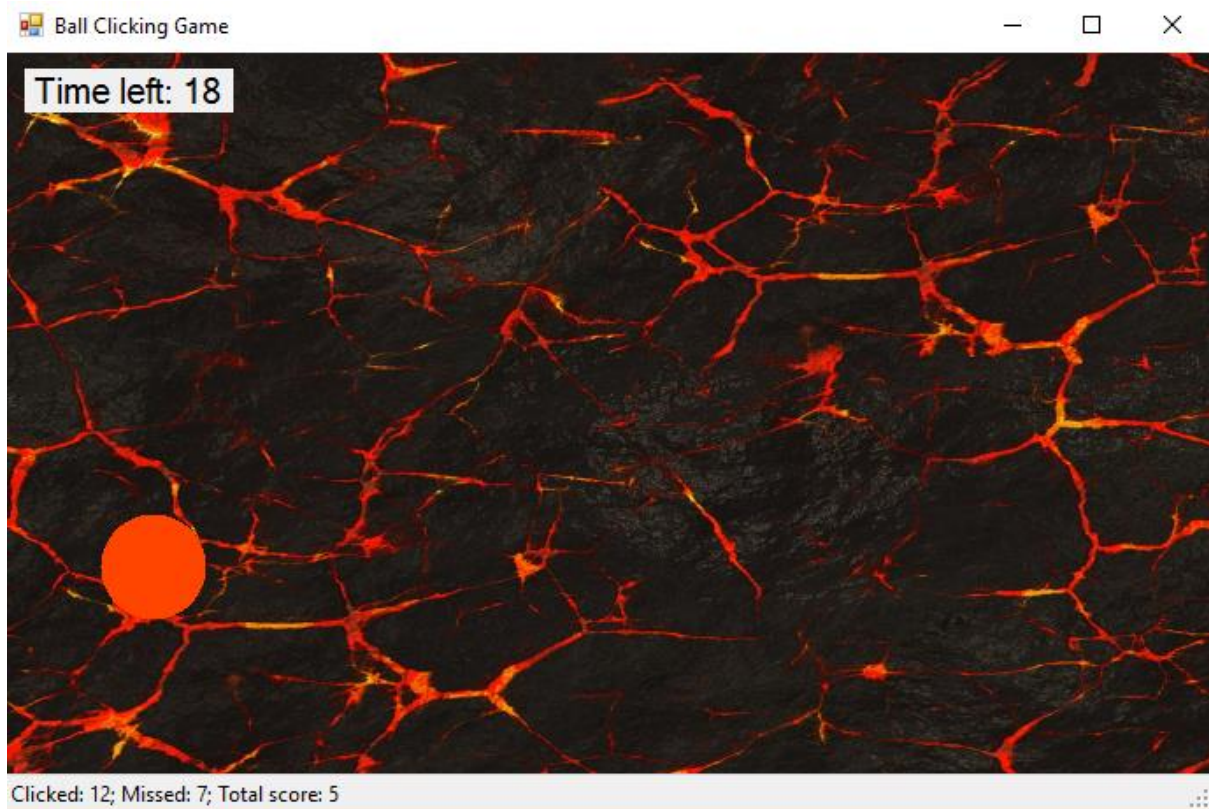


# Ball Clicking Game

## 1.Опис и цел на апликацијата:

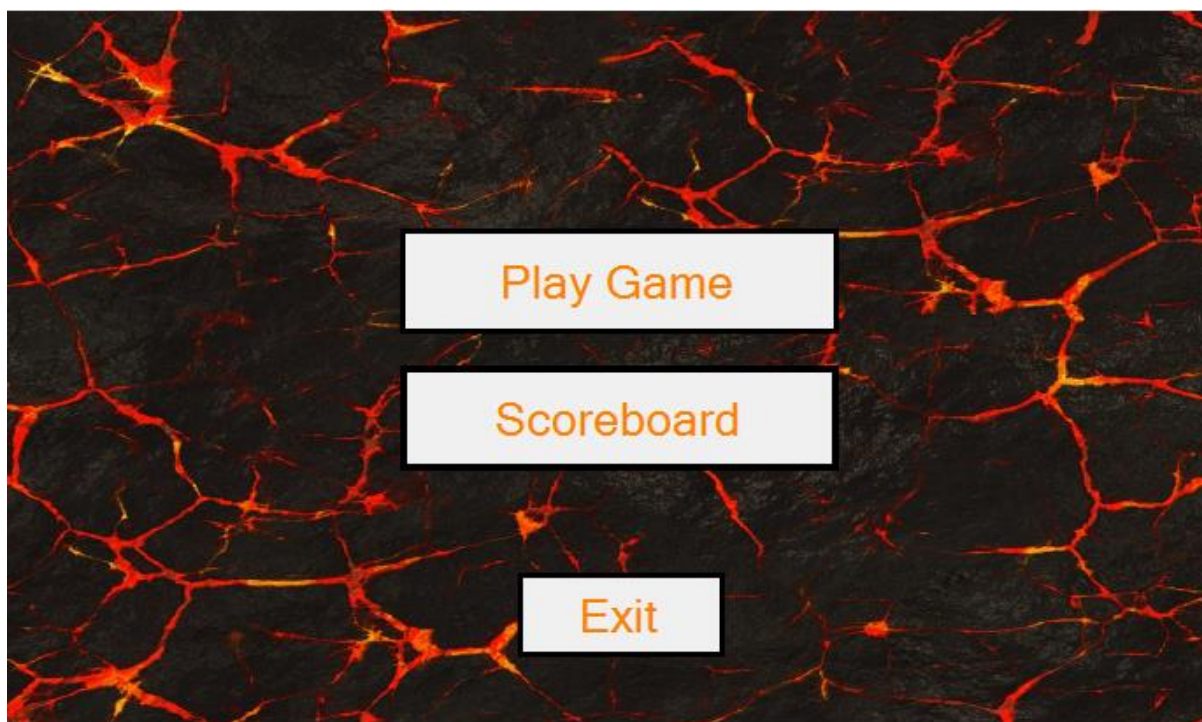
Апликацијата е едноставна игра која се игра со кликање на топче што се движи низ екранот за време од 30 секунди. Целта е да соберете што е можно повеќе поени. Поените се пресметуваат како бројот на кликувања на топчето одземени од бројот на промашувања на топчето.



## 2.Упатство за користење

### 2.1 Play game

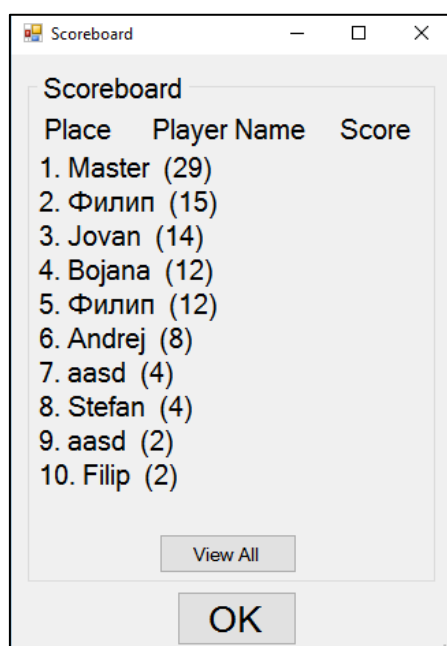
На почетниот прозорец(Слика 1) имаме можност да избереме почеток на играта(Play Game) или да ја видиме листата со поени (Scoreboard).



Слика 1

## 2.2 Scoreboard (Слика 2)

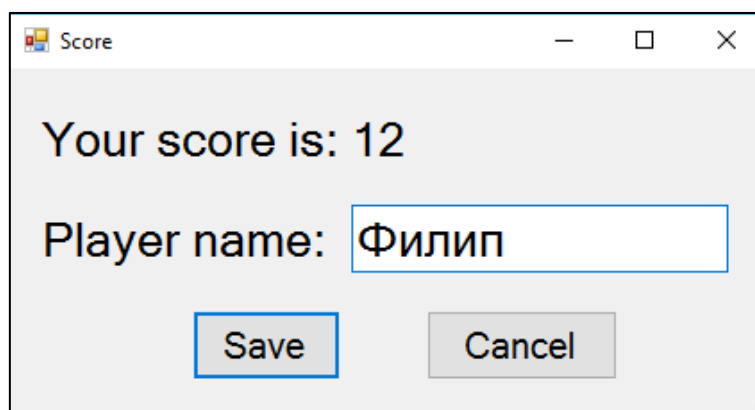
При клик на копчето Scoreboard првично се прикажуваат 10те најдобри играчи според бројот на постигнати поени, постои опција да се видат и останатите играчи најдобри 1000 играчи со притискање на копчето View All.



Слика 2

## 2.3 Save score (Слика 3)

При изминување на времето од 30 секунди се генерира прозорец кој ви нуди можност да го зачувате вашиот резултат со ваше име кое мора да биде поголемо од 3 и помало од 25 карактери со притискање на копчето Save од формата. Имињата може се повторуваат, односно еден играч.



Слика 3

Податоците се **сериализирани** и се достапни и послам исклучување на играта. Исто така постои можност за користење на База на податоци со користење на Entity Framework при што се користи Repository pattern. Кодот за функционалност со База на податоци е имплементиран и коментиран, но потребно е да има инсталирано SQLServer на вашата машина.

## 3. Претставување на проблемот

Целосното решение на проблемот е представено на Слика 4

### 3.1 Податочни структури

#### 3.1.1 Модели (класи)

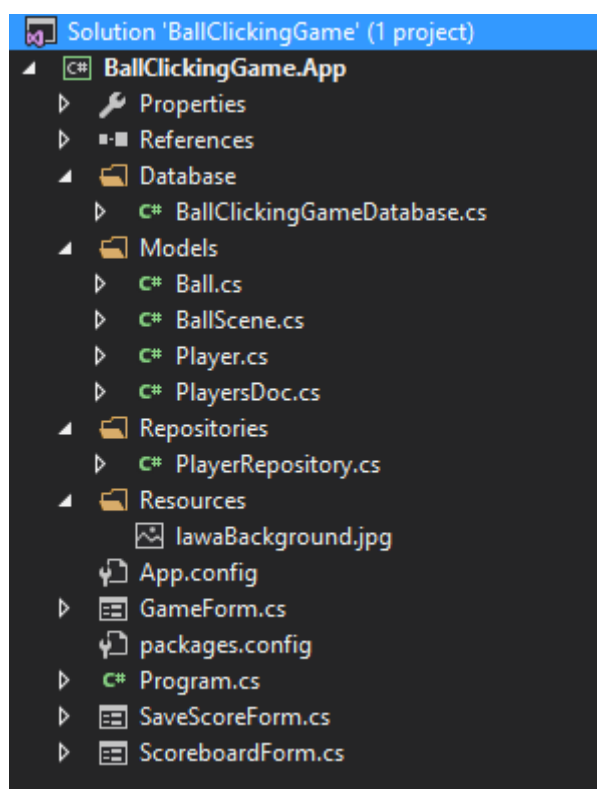
Податоците и функциите за топчето се чуваат во класата **Ball**, додека топчето е представено на сцена која е имплементирана во класата **BallScene**. Во класата **Player** се чуваат информации за податоци и функции кои се користат при сериализација и запишување на објектот во базата на податоци, додека класата **PlayersDoc** содржи функционалност која се користи при сериализација на документот, односно Scoreboard листата

и манипулација со играчите за време на користење на апликацијата.

Базата на податоци со користење на **EntityFramework** е имплементирана во класата **BallClickingGameDatabase**, додека објект од неа се креира и со неа се манипулира во складиштето на играчи(Players repository).

### 3.1.2 Форми

Во апликацијата се користат три форми, **GameForm** за главна манипулацијата и две помошни форми **SaveScoreForm** и **ScoreboardForm** за зачувување и приказ на играчите и нивните поени.



Слика 4

## 4.Некои методи, алгоритми и функционалности

**Move()** - метода за движење на топчето низ формата со одбивање од рабовите на истата. Слика 5

**GetTop10Players()** – метода за сортирање и земање на најдобрите 10 играчи по опаѓачки редослед користејќи ја LINQ Lambda expression синтаксата. Слика 7. Некои пропертија во класите се документиран со **<summary> xml** тагови за полесно користење на класите во нивна употреба при понатамошна изработка на проектот. Слика 6

```
//Moving the ball around the scene
public void Move()
{
    if ((Position.X - Radius < Math.Abs(dx)) ||
        (Position.X + (Radius * 2) + Math.Abs(dx)) > SceneWidth + Math.Abs(dx))
    {
        dx = -dx;
    }
    if ((Position.Y < 80) ||
        (Position.Y + (Radius * 2) + Math.Abs(dy)) > SceneHeight - Math.Abs(dy) - 22)
    {
        dy = -dy;
    }
    Position = new Point(Position.X + dx, Position.Y + dy);
}
```

Слика 5

```
/// <summary>
///We need ID to save and search the player in database
/// </summary>
public int ID { get; set; }
/// <summary>
/// Is player active in database,
/// if the player is active in database he will be visible in the scoreboard
/// </summary>
public bool IsActive { get; set; }
```

Слика 6

```
//Get top 10 players by descending
public List<Player> getTop10Players()
{
    return Players.Where(player => player.IsActive == true)
        .OrderByDescending(player => player.getScore())
        .Take(10).ToList();
}
```

Слика 7

Апликацијата и класите се отворени за проширување (Reusable) при понатамошно развивање на играта, односно би можело да се прошири апликацијата за играње во повеќе модови на пример мод со custom time на играње, повеќе топчиња кои би се движеле по екранот, секое од тие топчиња би носело различен број на поени и би се движело со различна брзина со креирање на нови класи и наследување од основите досега креирани класи при стандардниот едноставен мод.

Проектот е достапен на: Dropbox -

<https://www.dropbox.com/sh/52m5v8bjz2g4y23/AACcpLvQ1eiUfkwoz4o3Qy3Ya?dl=0>

GitHub - <https://github.com/fpipe/BallClickingGame>