# Multimedia

# HTML 5 GAME

**Team Members :**

| | |
|---|---|
| **Mohab Zearban** | **1206** |
| **Mohamed El Masry** | **1090** |

**Game Name :** TANKZ

***Game Description :-***

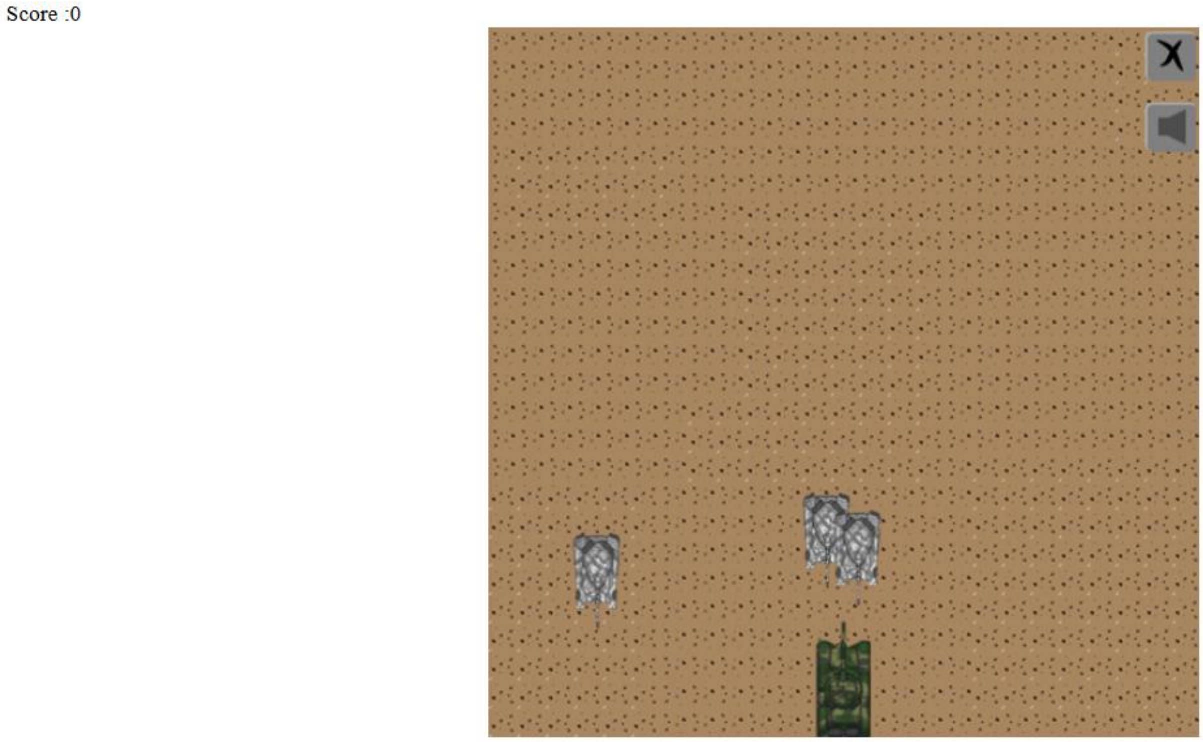
**T**he Player is the Green Tank Defending Its Line .
**M**ovement only in Left and Right Directions using Arrow Keys.
**P**layer Can Look in Three Directions (Up,Left and Right).
**T**he Player Have Only one Bullet which Regenerates after it reaches the End of the Canvas.
**T**he Bullet Goes Through all Enemies in its way.
**T**he Enemies are the Grey Tanks.
**W**hen the Player Score Reaches (100) Player Moves to the Next Level.
**D**ifficulity of the Game Depends on two Variables the Speed of the Enemy and the Number of Enemies.
**T**he Player Has Only One Live after it a Game Over and Returns to Menu.
**T**he Game has Only 2 Levels.

## ***Technologies Used :-***

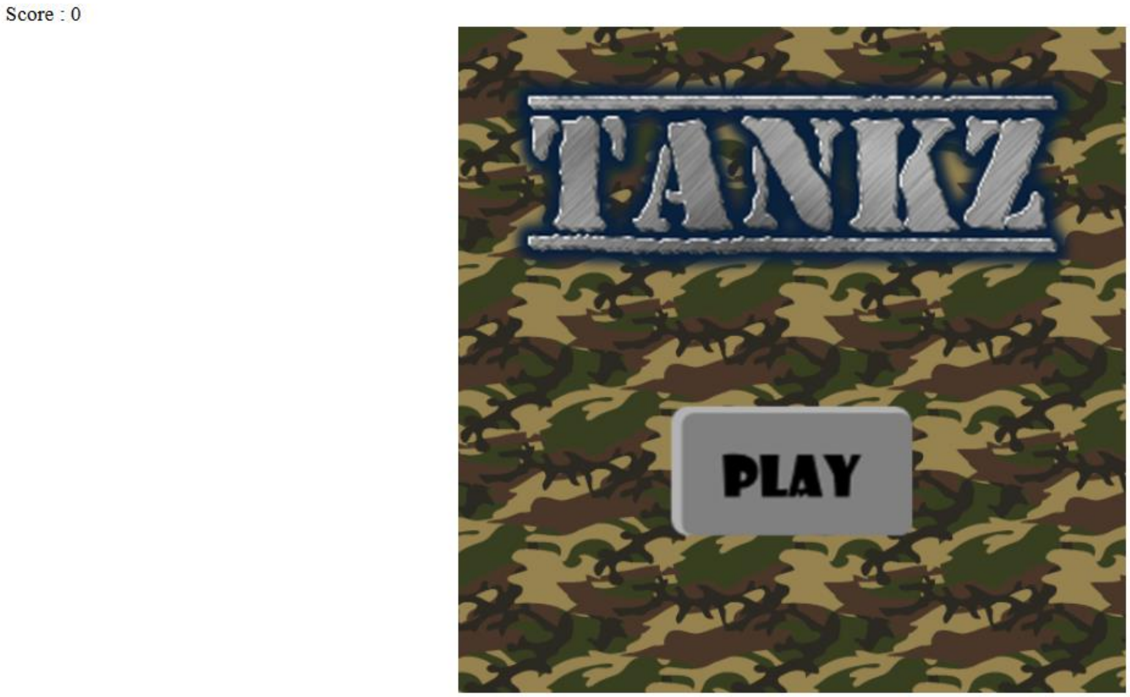
JavaScript (Scripting Language)
HTML 5 (Markup Language)
PIXI.js (Rendering Engine for HTML 5)

## ***SnapShots :-***

Mozilla FireFox – Windows

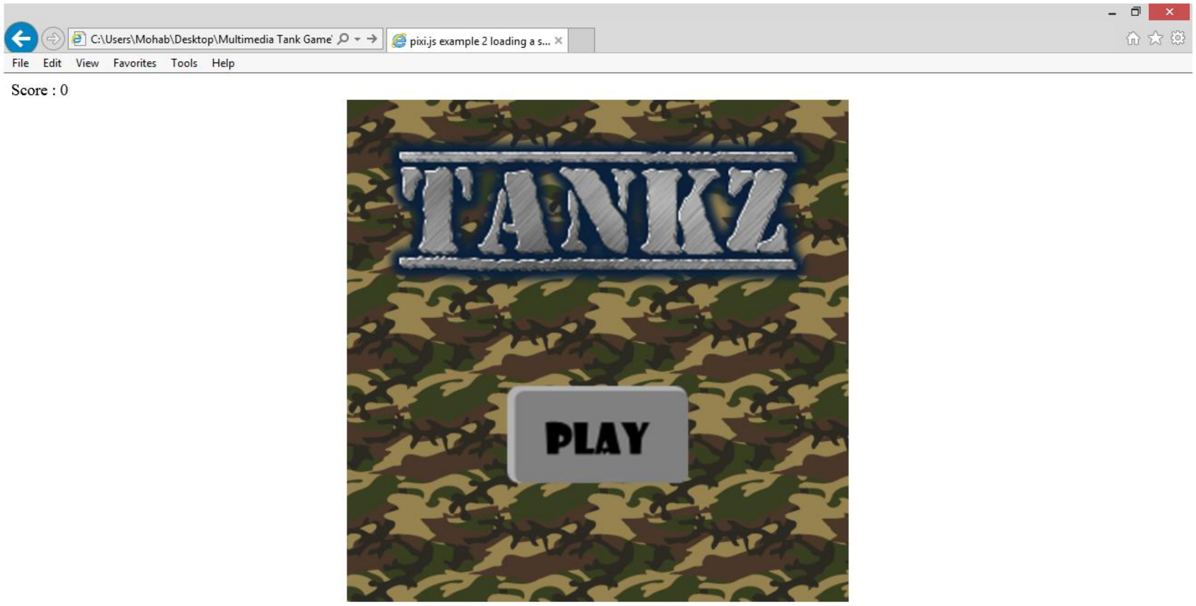


Score :0



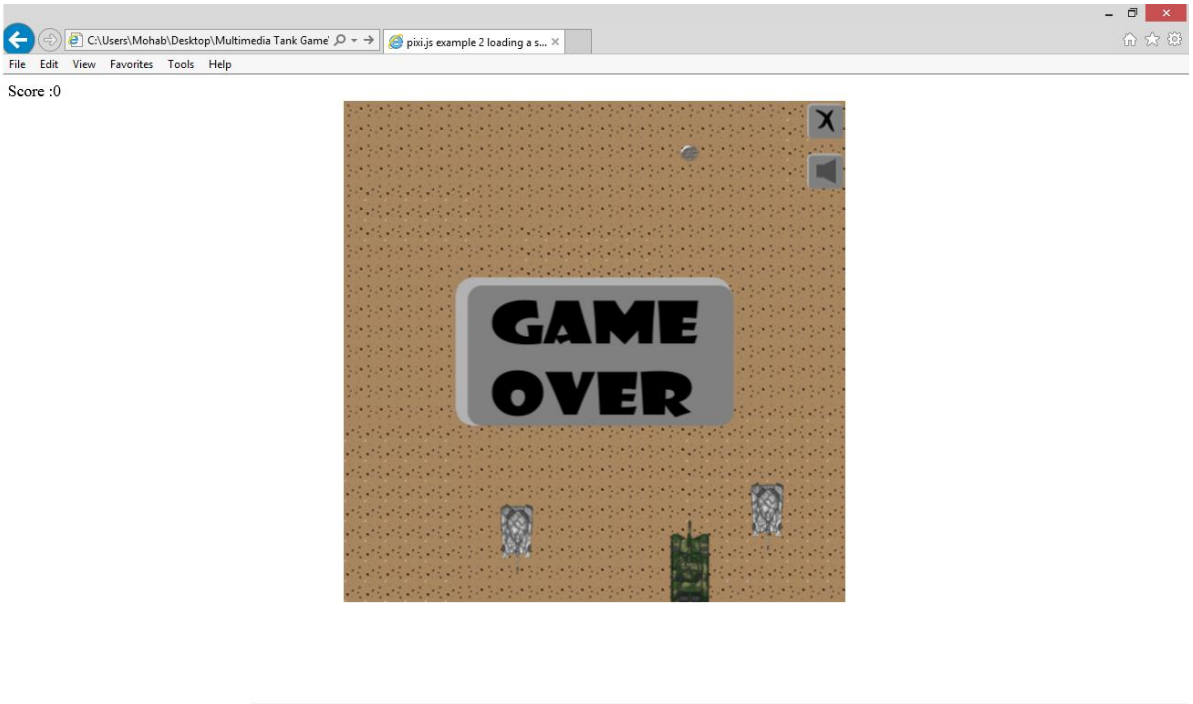
Score : 0

# Internet Explorer – Windows



C:\Users\Mohab\Desktop\Multimedia Tank Game'
pixi.js example 2 loading a s...
File
Edit
View
Favorites
Tools
Help
TANKZ
PLAY

Score : 0


C:\Users\Mohab\Desktop\Multimedia Tank Game'
pixi.js example 2 loading a s...
File Edit View Favorites Tools Help
Score :0
GAME OVER

## ***Code Snippets :-***

<u>-The Settings Function</u>

```
function Settings ()
{
            Exit.position.x = 480;
            Exit.position.y = 20;
            Exit.anchor.x = 0.5;
            Exit.anchor.y = 0.5;
            stage.addChild(Exit);


            SoundButton.position.x = 480;
            SoundButton.position.y = 70;
            SoundButton.anchor.x = 0.5;
            SoundButton.anchor.y = 0.5;
            stage.addChild(SoundButton);

            // set the mousedown and touchstart callback..
            Exit.mousedown = function(data){
                  this.alpha = 1;
                  GameMode = 0;
                  GameLevel = 1;
                  DeleteLevel ();
```

```
                stage.removeChild(ExitButton);
            }

          // set the mousedown and touchstart callback..
            SoundButton.mousedown = function(data){

                if(!Mute){this.setTexture(SoundOff);Mute =
true;bgm.pause();}else{this.setTexture(SoundOn);Mute = false;bgm.play();}
                this.alpha = 1;
                // Mute Sound and Stop BGM



                stage.removeChild(SoundButton);
            }


    }
```

-Game Update and Game Loop Functions the main Functions Running the game :-

```
function GameUpdate ()
    {


        if(GameMode == GAME_MODE.MENU)
        {

            DrawMenu();

            // set the mousedown and touchstart callback..
            PlayButton.mousedown = function(data){

                this.setTexture(PlayButtonDown);
                this.alpha = 1;
                GameMode = 1;
                GameLevel = 1;
                stage.removeChild(MenuBG);
                stage.removeChild(PlayButton);
                InitLevel(3,3);
            }

            // set the mouseover callback..
            PlayButton.mouseover = function(data){

                this.setTexture(PlayButtonDown);
            }

            // set the mouseout callback..
            PlayButton.mouseout = function(data){

                this.setTexture(PlayButtonUp)
            }

            // set the mouseup and touchend callback..
            PlayButton.mouseup = PlayButton.touchend = function(data){
            //this.isdown = false;

                this.setTexture(PlayButtonUp);

            }
        }

        else if (GameMode == GAME_MODE.PLAYING)
        {
            //Draw Level
            stage.addChild(LevelBG);
            ShowScore (Score);
            Settings ();


            //Update Enemies
            UpdataEnemies ();
            DrawEnemies ();

            //Update Player

            if(key[37])
            {
                //Move Left
                Player.position.x -= MovSpeed;
                Player.rotation = ((Math.PI/180) * (-1*90) );
            }
            else if(key[39])
            {
                //Move Right
                Player.position.x += MovSpeed;
                Player.rotation = ((Math.PI/180) * (90) );;
            }
            else if (key[38])
            {
                //Look Up
                Player.rotation = 0;
            }
```

```
            if (key[32] && !BulletAlive)
            {
                //Shoot At Enemy
                BulletAlive = true;
                if(!Mute){ShootSnd.play();}
                //Init Bullet to Player Position
                PlayerBullet.position.x = Player.position.x;
                PlayerBullet.position.y = Player.position.y - (Player.height / 2);
                //Look At Enemy
                Player.rotation = 0;
            }

            //Click P to Pause
            if (key[80])
            {
                GameMode = 3;
            }

            stage.addChild(Player);

            //Explode (Player);

            if (BulletAlive) UpdatePlayerBullet();

            //Check Wining Condition
            if(Score >= 100) {if(GameLevel == 1){GameMode = 4 ;}else if(GameLevel == 2)
{GameMode = 2}}


        }
        else if (GameMode == GAME_MODE.PAUSE)
        {

            PauseLbl.position.x = 250;
            PauseLbl.position.y = 250;
            PauseLbl.anchor.x = 0.5;
            PauseLbl.anchor.y = 0.5;
            stage.addChild(PauseLbl);

            //Click Esc to Continue
            if (key[27])
            {
                GameMode = 1;
                stage.removeChild(PauseLbl);
                console.log("aloo");
            }
```

```
}
else if  (GameMode == GAME_MODE.GAMEOVER )
{
        GameOverButton.position.x = 250;
        GameOverButton.position.y = 250;
        GameOverButton.anchor.x = 0.5;
        GameOverButton.anchor.y = 0.5;
        stage.addChild(GameOverButton);

        // set the mousedown and touchstart callback..
        GameOverButton.mousedown = function(data){

                this.setTexture(GameOverOn);
                this.alpha = 1;
                GameMode = 0;
                DeleteLevel ();
                stage.removeChild(GameOverButton);
        }

        // set the mouseover callback..
        GameOverButton.mouseover = function(data){

                this.setTexture(GameOverOn);
        }

        // set the mouseout callback..
        GameOverButton.mouseout = function(data){

                this.setTexture(GameOverOff);
        }

        // set the mouseup and touchend callback..
        GameOverButton.mouseup = GameOverButton.touchend = function(data){
        //this.isdown = false;

                this.setTexture(GameOverOff);

        }

}
else if (GameMode == GAME_MODE.WON )
{
        NextLevelButton.position.x = 250;
        NextLevelButton.position.y = 250;
        NextLevelButton.anchor.x = 0.5;
        NextLevelButton.anchor.y = 0.5;
        stage.addChild(NextLevelButton);
```

```
                        // set the mousedown and touchstart callback..
                        NextLevelButton.mousedown = function(data){

                                this.setTexture(NextLevelOn);
                                this.alpha = 1;
                                GameMode = 1;
                                GameLevel = 2;
                                DeleteLevel ();
                                InitLevel (5,5);
                                stage.removeChild(NextLevelButton);
                        }

                        // set the mouseover callback..
                        NextLevelButton.mouseover = function(data){

                                this.setTexture(NextLevelOn);
                        }

                        // set the mouseout callback..
                        GameOverButton.mouseout = function(data){

                                this.setTexture(NextLevelOff);
                        }

                        // set the mouseup and touchend callback..
                        NextLevelButton.mouseup = NextLevelButton.touchend = function(data){
                        //this.isdown = false;

                                this.setTexture(NextLevelOff);

                        }
                }
        }




        function GameLoop() {

                GameUpdate();
          requestAnimFrame( GameLoop );

          renderer.render(stage);
        }
```

-Level Initialization Function which can be used For Multiple Levels :-

```
function InitLevel (EnemyNumber, ESpeed)
        {
                NoEnemies = EnemyNumber;
                EnemySpeed = ESpeed;
                LevelBG = new PIXI.Sprite.fromFrame("TankBackGround.png");
                var PlayerTex = new PIXI.Texture.fromFrame("Player.png");
                Player = new PIXI.Sprite(PlayerTex);
                var PlayerBulletTex = new PIXI.Texture.fromFrame("PlayerBullet.png");
                PlayerBullet = new PIXI.Sprite(PlayerBulletTex);
                var EnemyTex = new PIXI.Texture.fromFrame("Enemy.png");
                var PauseLblTex = new PIXI.Texture.fromFrame("Pause.png");
                PauseLbl = new PIXI.Sprite(PauseLblTex);
                GameOverOn = new PIXI.Texture.fromFrame("GameOver2.png");
                GameOverOff = new PIXI.Texture.fromFrame("GameOver1.png");
                GameOverButton  = new PIXI.Sprite(GameOverOff);
                var ExitTex = new PIXI.Texture.fromFrame("Exit.png");
                Exit = new PIXI.Sprite(ExitTex);
                SoundOn = new PIXI.Texture.fromFrame("SoundOn.png");
                SoundOff = new PIXI.Texture.fromFrame("SoundOff.png");
                SoundButton = new PIXI.Sprite(SoundOn);
                NextLevelOn = new PIXI.Texture.fromFrame("NextLevel2.png");
                NextLevelOff = new PIXI.Texture.fromFrame("NextLevel1.png");
                NextLevelButton  = new PIXI.Sprite(NextLevelOff);
                GameOverButton.setInteractive(true);
                SoundButton.setInteractive(true);
                Exit.setInteractive(true);
                NextLevelButton.setInteractive(true);

                Score = 0;

                for ( var i=0 ; i < NoEnemies ; i++ )
                {
                        Enemies[i] = new PIXI.Sprite(EnemyTex);
                        Enemies[i].anchor.x = 0.5;
                        Enemies[i].anchor.y = 0.5;
                }

                //Initialize Explosion MovieClip
                for (var i=0; i < 26; i++)
                        {
                        var texture = PIXI.Texture.fromFrame("Explosion_Sequence_A " + (i+1) +
".png");
                        explosionTextures.push(texture);
                        };


                InitEnemyPositions ();
                Player.anchor.x = 0.5;
                Player.anchor.y = 0.5;
                Player.position.x = 250;
                Player.position.y = renderer.height - 40;
                Player.rotation = 0;
                PlayerBullet.anchor.x = 0.5;
                PlayerBullet.anchor.y = 0.5;




        }
```