

## Welcome to creating mini games with 3D Game studio A8.

Written by : René Pol 04-08-2017©

As you might have noticed the hero does not move diagonal. If you press key up and left it does not work. We will improve the movement by giving it its own variable and function.

```
var is_input_pressed()
{
if(key_cul != 0 || key_cuu != 0 || key_cur != 0 || key_cud != 0)
{
return 1;
}
return 0;
}
```

```
function movement_input(ENTITY* my)
{
if(is_input_pressed())
{
var side_direction = 0;
var up_direction = 0;
if(key_cul){side_direction = 1;}
if(key_cur){side_direction = 2;}
if(key_cuu){up_direction = 1;}
if(key_cud){up_direction = 2;}
//finalizing angles
//horizontal movement
if(up_direction == 0)
{
if(side_direction == 1)
{
my.pan = 90;
}
if(side_direction == 2)
{
my.pan = 270;
}
}
//vertical movement
if(side_direction == 0)
{
if(up_direction == 1)
{
my.pan = 0;
}
if(up_direction == 2)
{
my.pan = 180;
}
}
//diagonal movement
if(up_direction != 0 && side_direction != 0)
{
if(side_direction == 1 && up_direction == 1)
{
my.pan = 45;
}
if(side_direction == 2 && up_direction == 1)
{
my.pan = 315;
}
if(side_direction == 1 && up_direction == 2)
{
my.pan = 135;
}
if(side_direction == 2 && up_direction == 2)
{
my.pan = 225;
}
}
}
```

## Welcome to creating mini games with 3D Game studio A8.

Written by : René Pol 04-08-2017©

In order to make it all work we need to insert the function movement\_input into the walking\_sid action.

```
action walking_sid()
{
set(my,SHADOW |POLYGON);
sid = me;
c_setminmax(me);
var standing_percentage;
var walking_percentage;
var jumping_percentage;
on_space = my_jump;
my.skill1 =200;
my.skill20 =0;
my.skill21=0;
while(1)
{
my_gravity();
movement_input(my);
if(sid.z < -21){
set(my,INVISIBLE |PASSABLE);
my.event = NULL;
wait(1);
snd_play(no_snd,100,0);
wait(-2);
ent_create("sid.mdl", vector(1.597,0, 19.507),walking_sid);
wait(1);
ent_remove(me);
break;
}

while(key_space)
{
if(jumping ==0){
snd_play(jump_snd,100,0);
jumping =1;
}
ent_animate(my, "jump", jumping_percentage, 0);
jumping_percentage += 4 * time_step;
c_move (my, vector (10 * time_step, 0, absdist.z), nullvector, IGNORE_PASSABLE | GLIDE | IGNORE_SPRITES);
wait(1);
}
result = c_move(me, nullvector, vector(10 * (key_cuu - key_cud) * time_step, 10 * (key_cul - key_cur) * time_step,absdist.z), GLIDE |
IGNORE_PASSABLE | IGNORE_SPRITES | ENABLE_BLOCK);
if(result > 0)
{
walking_percentage %= 100;
walking_percentage += 15 * time_step;
ent_animate(my, "run", walking_percentage, ANM_CYCLE);
}
else{
standing_percentage %= 100;
standing_percentage += 5 * time_step;
ent_animate(my, "stand", standing_percentage, ANM_CYCLE);
}
wait(1);
}
}
```

Yes ! Now Sid moves diagonal as well. You can use all directions you need for any kind of game.  
Always handy to have this script.

Until now we used a simple camera (position) that is placed in the level. Let have some fun with different use of camera settings shall we ?

We begin with a camera that sticks to one position but will always follow the players character. For this we create a camera function.

```
////////////////////////////////////following camera////////////////////////////////////  
  
function follow_camera() // this fixed camera changes its angles, focusing on the player, camera_type = 6  
{  
  VECTOR temp;  
  while (!sid)  
  {  
    wait (1);  
  }  
  camera.arc = 60; vec_set (camera.x, vector(-200, 200, 350));  
  while (1)  
  {  
    vec_set(temp.x, sid.x);  
    vec_sub(temp.x, camera.x);  
    vec_to_angle(camera.pan, temp); // rotate the camera toward the player every frame  
    wait (1);  
  }  
}
```

We call this function in the walking\_sid movement.

```
while(1)  
{  
  follow_camera();  
  my_gravity();  
  movement_input(my);  
}
```

You can place this camera where ever you want by changing its coordinates.



## Welcome to creating mini games with 3D Game studio A8.

Written by : René Pol 04-08-2017©

Instead of making the camera follow the main character we can also use a top view camera that oversees the complete level.

We create a camera function for this and activate it like we did before.

```
function top_camera()
{
while (!sid) {wait (1);} // wait until the player is created
camera.tilt = -90; // the camera will look downwards all the time
while (1)
{
vec_set (camera.x, sid.x); //the player and the camera have the same x and y position
camera.z = sid.z + 500; // but the camera is placed 500 quants higher; play with this value
wait (1);
}
}
```

We call this function in the walking\_sid movement.

```
while(1)
{
top_camera();
my_gravity();
movement_input(my);
}
```

Voila a top view game is created. That was all pretty simple 😊

I created some more camera views you can try out yourself.

```
isometric_camera()
function firstperson_camera()
```

Use the one that is best for your game. I think the thirdperson\_camera is the right one for this little game.

## Welcome to creating mini games with 3D Game studio A8.

Written by : René Pol 04-08-2017©

One more thing to finalize this first mini game we made. We want a logo on start up. This is all pretty simple. Here is how it works.

You create 2 graphics. 1 black and one with your logo or game title on it. On game begin (main function) it will start with the black screen with the logo screen under it. The black screen will fade out so the logo is shown. Then the black fades in again and the logo screen is removed. When the black screen fades out again it shows the game and the black screen gets removed.

This is how it all works in the main function.

```
//////////////////////////////////main function//////////////////////////////////
```

```
function main()
{
video_set(1024,768,32,0);
set(black_pan,SHOW | TRANSLUCENT);
set(logo_pan,SHOW | TRANSLUCENT);
black_pan.alpha = 100;
wait(-2);
soundtrack01_handle = media_loop("ingame.mp3",NULL,100);
while(black_pan.alpha >0){
black_pan.alpha -=3*time_step;
wait(1);
}
wait(-2);
while(black_pan.alpha <100){
black_pan.alpha +=3*time_step;
wait(1);
}
wait(-1);
reset(logo_pan,SHOW);
sky_color.red = 0;
sky_color.green = 0;
sky_color.blue = 0;
while(black_pan.alpha >0){
black_pan.alpha -=3*time_step;
wait(1);
}
level_load("BMS.WMB");
mouse_map = pointer_png;
mouse_mode = 3;
set(menu_pan,SHOW);
set(timerpanel_pan,SHOW);
set(pan_score,SHOW);
set(timer_pan,SHOW);
on_p = pause;
reset(black_pan,SHOW | TRANSLUCENT);
}
```

While I was writing this doc I came up with another thing to add to the game. A simple highscore.

## Welcome to creating mini games with 3D Game studio A8.

Written by : René Pol 04-08-2017©

Like the normal score we create a variable for the highscore and make it visible.

```
var hscore =0;

FONT* fnt1_pan = "Supercell-Magic#28b";
PANEL* pan_hscore = {digits=700,50,"Highscore : %006.0f",fnt1_pan,1,hscore;
layer = 10;
flags = SHOW;green=255; blue=0; red=255;
}
```

We create a function that checks if a highscore is broken so a sound is played.

```
////////////////////////////////////function check highscore////////////////////////////////////
var play_sound = 0;
function check_highscore()
{

if(score > hscore && play_sound ==0){
play_sound =1;
snd_play(hscore_snd,100,0);
}
}
```

Every time a star is picked up this function is called.

```
action stars_pick()
{
set(my,BRIGHT | METAL | PASSABLE | SHADOW);
var RandomX = random(213);
var RandomY = random(155);
while(!sid){
wait(1);
}
while(1)
{
my.pan -=5*time_step;
wait(1);
if (vec_dist (sid.x, my.x) < 20){
set(my,INVISIBLE);
my.event = NULL;
wait(1);
score +=10;
snd_play(stars_snd,100,0);
ent_remove(me);
check_highscore();
wait(-2);
ent_create("star.mdl", vector(RandomX, RandomY, 44),stars_pick);
break;
}
}
}
```

If you want the highscore to show once the game is over we need to add it into the game finish script.

```
function game_finish()
{
if(score > hscore){
hscore = score;
play_sound = 0;
}
media_stop(soundtrack01_handle);
set(finish_pan,SHOW);
snd_play(finish_snd,100,0);
set(sid,PASSABLE | INVISIBLE);
sid.event = NULL;
wait(-1);
snd_play(finito_snd,100,0);
ent_remove(sid);
wait(-1);
reset(finish_pan,SHOW);
game_begin = 0;
wait(-2.5);
soundtrack01_handle = media_loop("ingame.mp3",NULL,100);
set(menu_pan,SHOW);
mouse_mode = 3;
}
```



## Welcome to creating mini games with 3D Game studio A8.

Written by : René Pol 04-08-2017©

Now the last thing to do is make this highscore be saved when the game is quit. So when the game is loaded the highscore at that moment will show. Thanks to forum user Kartoffel we have a fun and easy way to do this. We create a load and save highscore function.

```
//////////////////////////////////load and saving highscore//////////////////////////////////

#define HIGH_SCORE_SAVE_FILE "hs.score"
TEXT* input_file = { strings = 1; } // used to load saved information, only one string needed
function load_highscore()
{
    txt_load(input_file, HIGH_SCORE_SAVE_FILE);
    wait(1); // wait one frame to load the text file - important!
    hscore = str_to_float( (input_file.pstring)[0] ); // (<text>.pstring) [ line, beginning with 0! ] -> convert to float
}

function save_highscore()
{
    var file_handle = file_open_write(HIGH_SCORE_SAVE_FILE);
    wait(1); // wait to open the file?? - not sure if this is needed
    file_str_write(file_handle, str_for_float(NULL, (double)hscore)); // convert the highscore to a string -> write into file
    file_close(file_handle);
}
```

In the main function we make sure the highscore gets loaded.

```
function main()
{
    video_set(1024,768,32,0);
    set(black_pan,SHOW | TRANSLUCENT);
    set(logo_pan,SHOW | TRANSLUCENT);
    black_pan.alpha = 100;
    wait(-2);
    soundtrack01_handle = media_loop("ingame.mp3",NULL,100);
    while(black_pan.alpha >0){
        black_pan.alpha -=3*time_step;
        wait(1);
    }
    wait(-2);
    while(black_pan.alpha <100){
        black_pan.alpha +=3*time_step;
        wait(1);
    }
    wait(-1);
    reset(logo_pan,SHOW);
    sky_color.red = 0;
    sky_color.green = 0;
    sky_color.blue = 0;
    while(black_pan.alpha >0){
        black_pan.alpha -=3*time_step;
        wait(1);
    }
    level_load("BMS.WMB");
    load_highscore();
    mouse_map = pointer_png;
    mouse_mode = 3;
    set(menu_pan,SHOW);
    set(timerpanel_pan,SHOW);
    set(pan_score,SHOW);
    set(timer_pan,SHOW);
    set(pan_hscore,SHOW);
    on_p = pause;
    reset(black_pan,SHOW | TRANSLUCENT);
}
```



Now remember were we made sure to set the highscore ? Well lets save it there also.

```
function game_finish()
{
if(score > hscore){
hscore = score;
play_sound = 0;
save_highscore();
}
media_stop(soundtrack01_handle);
set(finish_pan,SHOW);
snd_play(finish_snd,100,0);
set(sid,PASSABLE | INVISIBLE);
sid.event = NULL;
wait(-1);
snd_play(finito_snd,100,0);
ent_remove(sid);
wait(-1);
reset(finish_pan,SHOW);
game_begin =0;
wait(-2.5);
soundtrack01_handle = media_loop("ingame.mp3",NULL,100);
set(menu_pan,SHOW);
mouse_mode = 3;
}
```

So now your game has it all. This is the first of mini games in this series. Future issues will be complete basic mini games using the same scripts but in different ways. You can give this game your own look and feel I am curious what you can make out of it.

Until next issue Happy game creation !

Rene Pol aka Realspawn.

