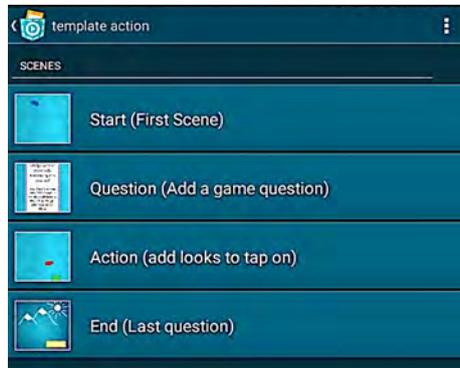


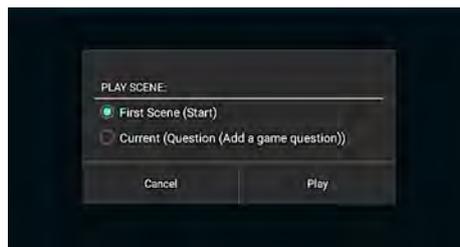
# Game Template Guide: Action game

The Action template contains 4 scenes:

- Start (First Scene)
- Question (Add a game question)
- Action (add looks to tap on)
- End (Last Scene)



Each of these scenes works like an individual program. For example, objects and variables can only be used for one scene. You can start a scene by tapping on the play button.



## Game-Play:

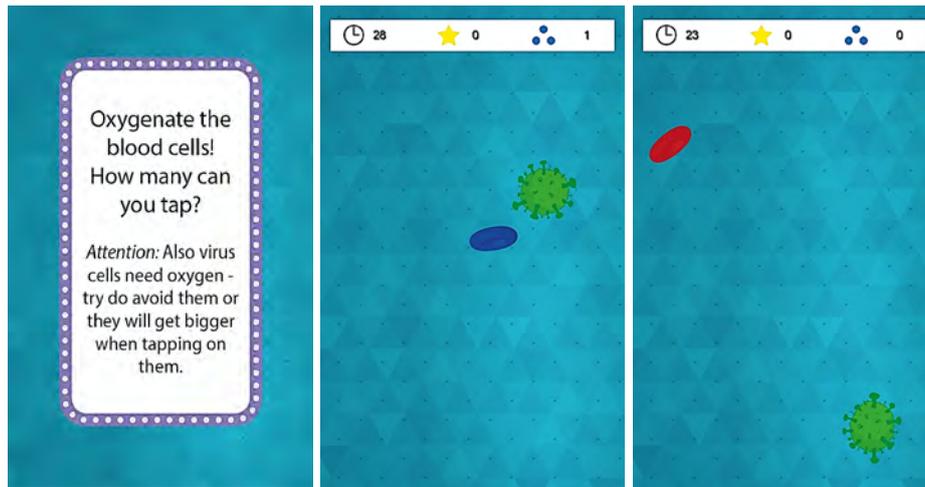
At the beginning of the game the user will get an info screen that tells him what he is allowed to tap on. During the game the different HUD-elements are shown – a timer, high score and the actual score display.

### **What is a HUD?**

A HUD is short for **H**eads **U**p **D**isplay and is a display area where gamers can view their characters vital statistics (such as health), and can indicate game progression (such as score). What attributes are shown in the HUD depends on the game.

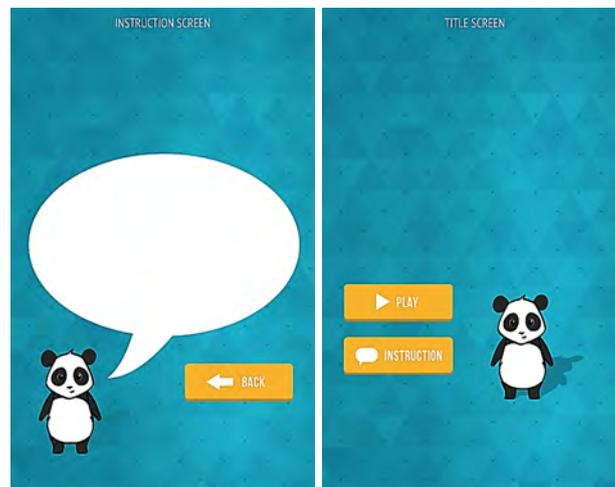
While the information that is displayed on the HUD depends greatly on the game, there are many features that players recognize across many games. Most of them are static onscreen so that they stay visible during gameplay.

You have 30 seconds (timer) to catch as many blood cells as you can. With each one you catch, the score increases by 1. The highscore is saved with each restart (restart button at the end of the game) and you can try to improve on this each time. Avoid the virus cells – if you catch one of these the virus cell increases in size and your score decreases by 1.



### Scene start

This scene contains the introduction of the Action game.



This scene contains the following objects:

- Background
- Initial game variables
- Group: start
  - Button play game
  - Button introduction
  - Button back

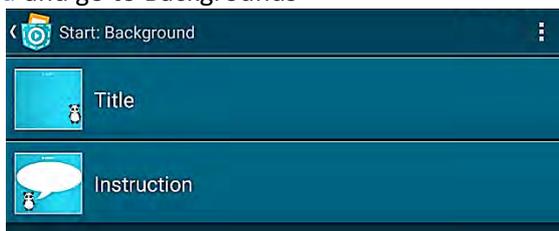


The object *Background* has two looks *Title* and *instruction* which can be changed and adapted.

**ToDo -1:**

Edit/change the looks of the object *Background*

1. Open *Background* and go to Backgrounds



2. Open the look *Title*: add the name or title of your action game.  
You can achieve this in two ways:
  - a. Add the name by using Pocket Paint's text tool
  - b. Delete this image and add your own which contains your title
3. Open the look *Instruction*: Add your instruction text
  - a. Add your text by using Pocket Paint's text tool
  - b. Delete this image and add your own which contains your text/instruction

The object *Initial Game Variables* contains the variables *score* and *hi-score*.

The group *Start* contains 3 objects (buttons):

- Object *Button Play game*: Starts the game – the game jumps to scene 2 *Question*
- Object *Button Instruction*: by tapping on this button you will get to the instruction screen
- Object *Button Back*: Press this button to go back to the title screen

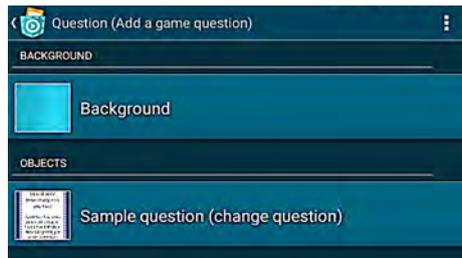
**Scene Question**

This scene presents a question for the game. Think about which elements the user has to tap on and which they have to avoid.

At the beginning of the game a question is defined that fits to the curriculum. If this is already mentioned in the instructions, and is not necessary, just delete this scene.

This scene contains the following objects:

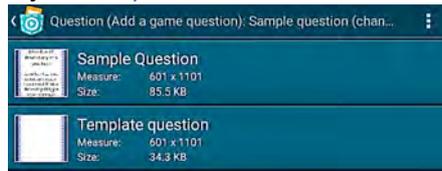
- Background
- Question



**ToDo - 2:**

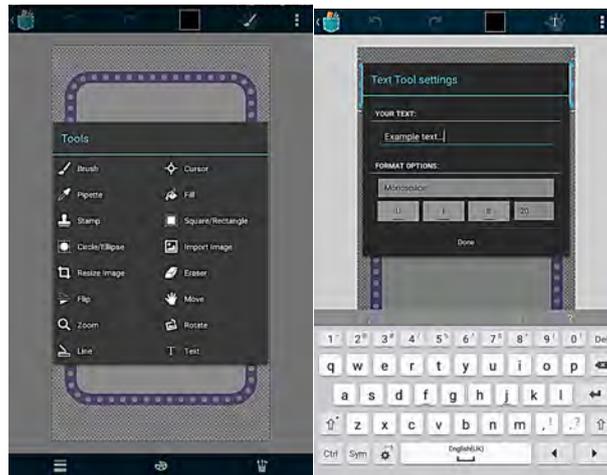
Change the question

1. Go to the looks of the object *Sample Question* and delete the look *Sample Question*



2. Edit the look *Template question* by using Pocket Paint and include text by using the text tool:

Pocket Paint



**Scene Action**

This scene contains the Action game from the topic Biology:Respiration - diffusion.  
Add your own objects here.

This scene contains the following objects:

- Background
- Group: Items
  - Item to tap
  - Item to avoid
- HUD Elements



The group *Items* contains the objects *Item to tap* and *Item to avoid*. In both objects local variables are defined *tapped correct* or *tapped wrong*. Local means that the variable is only true for this one object. This means that these objects can be copied and adapted as you like and have no influence on the other objects.

As long as the objects were not tapped the variable *tapped correct/wrong* is false and the objects move around on the screen and bounce off from the edge. If one object is tapped the variable *tapped correct/wrong* is true and a feedback will be sent (*look blood cell oxidised* or the virus cell's size gets bigger and the recording is played: *WaterDrop/beep*) At the same time the variable *score* is increased or decreased by 1. After tapping on one of the objects it hides and then appears at a random position.

The object *HUD-Elements* contains the variable *timer* that is set to 30 seconds and the other HUD elements. If the variable *timer* is 0 the scene *End* is shown.

Edit the scene *Action*

### ToDo - 3:

Add the looks that can be tapped.

1. Open the looks of your object *Item to tap*

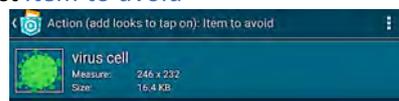


2. Delete both looks and create two new looks by tapping on the + symbol. The first look represents the *not-tapped* status and the other one represents the *tapped-on* status.
3. To add a new look you have different possibilities: Pocket Paint (Draw image), Media Library, Device or camera.



Add the looks that cannot be tapped.

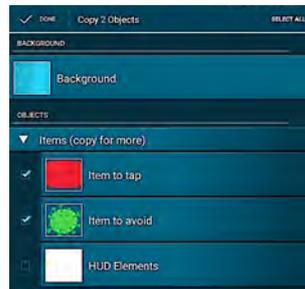
4. Open the looks of your object *Item to avoid*



5. Delete the look and create a new look by tapping on the + symbol. You need two different looks – The first look represents the *not tapped* status and the other one represents the *tapped on* status. The size of the look should increase when the object is tapped on.
6. This can be achieved by using code or by creating it in Pocket Paint.

Copy the objects *Item to tap* and *Item to avoid*

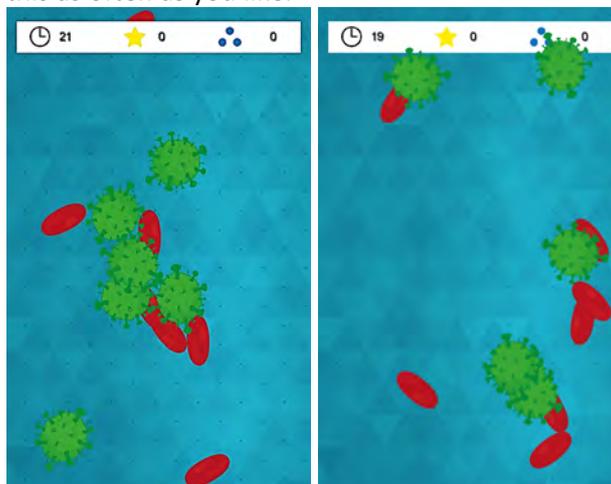
7. To increase the level of difficulty copy both looks by tapping on the overflow menu in the program overview.



8. Put these new objects into the group Items.



9. You can repeat this as often as you like.



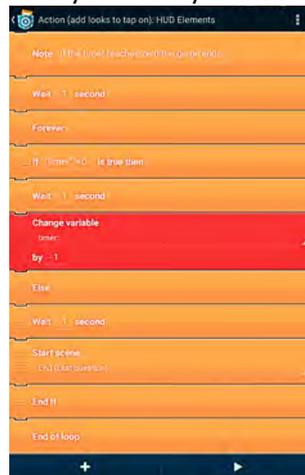
**ToDo-4:**

You can change the variable *timer* in the object *HUD Elements*, e.g. shorten or extend the time frame to play.

#### ToDo-5:

You can create similar action games. Those mini-games can be put into a sequence.

1. Copy the scene *Action*
2. Open the object *HUD Elements* in the scene *Action*.
3. Change the “Start scene” brick to your newly created scene after the timer ends.

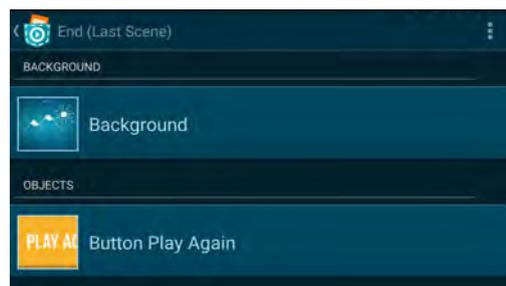


4. Attention: You need to create another score variable (*score 2*) that will be shown in your second scene. By creating another score, the score from scene 1 will not be overwritten. Both scores will be added up for creating the high score.

## Scene End

This scene contains the following objects:

- Background
- Button Play Again



Object *Button Play again* starts the second scene (Action). Additionally, if the variable *score* is greater than the variable *hi-score* the high score will be set to a new value. Afterwards the variable *score* is set to 0.

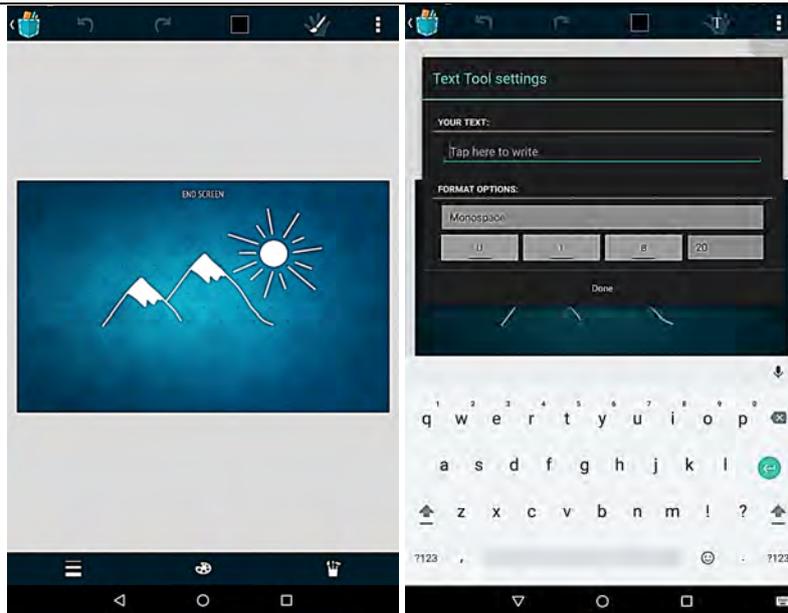
Edit the scene *End*

#### ToDo – 6:

Edit the End Screen

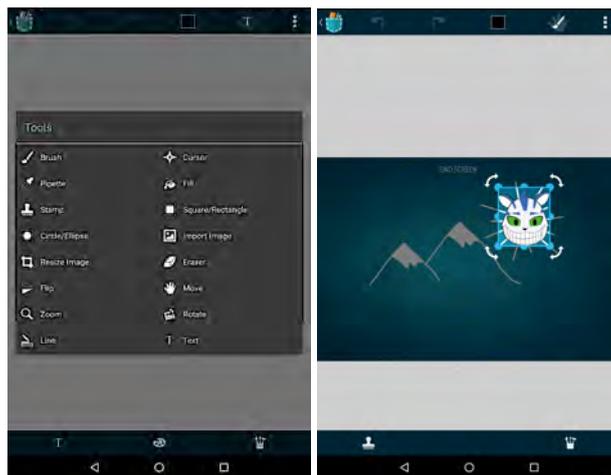
1. Open the object *Background* and go to its looks.
2. Edit the look *Background* by using Pocket Paint. You can include text by using the text tool:

Pocket Paint:



Additionally, you can also include pictures:

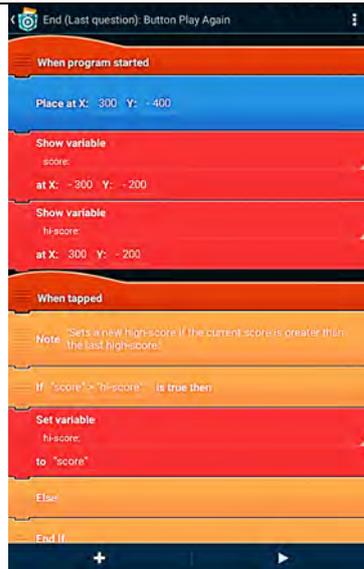
1. Tools (bottom right) – Import image



### ToDo - 7:

Display a score and high score

1. If you want you can use the data brick again to display both variables *score* and *hi score* at the end:



2. By using Pocket Paint you can add labels for both variables.