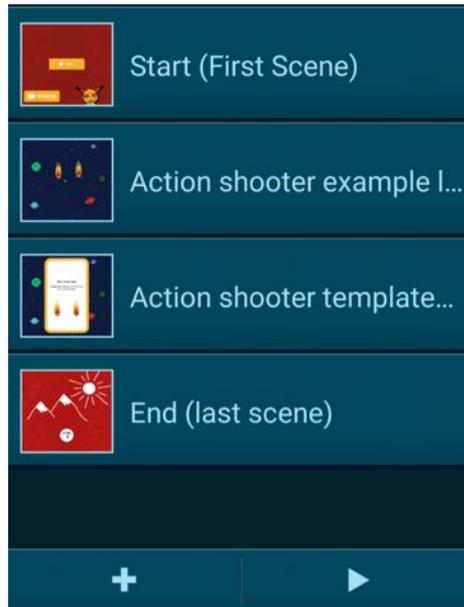


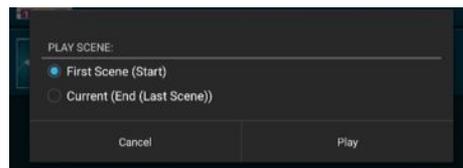
Game Template Guide: Action Shooter

The Action Shooter template has 4 scenes:

- Start (First scene)
- Action shooter example
- Action shooter template
- End (last scene)

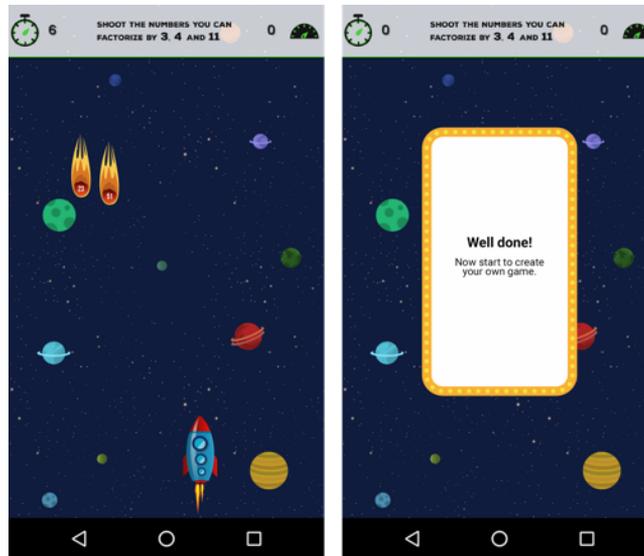


Each of these scenes work like an individual program. For example, objects and variables can only be used for one scene. You can start a scene by tapping on the play button.



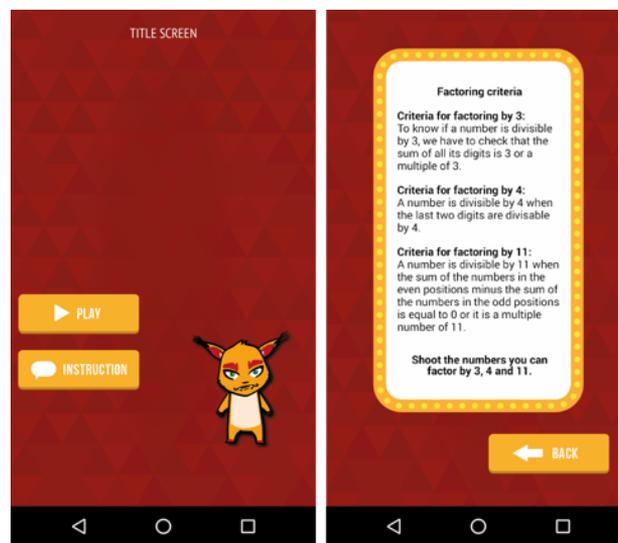
Gameplay:

The main goal of the **Action shooter template** is to shoot comets with numbers written on them. The spaceship shoots if it is tapped. The user has to control the movement of the spaceship by tilting the device left and right. The score is increased by one for the comet with the number divisible by 3, 4, and 11 without remainder. The score is decreased by one for the wrong shot (the number on the comet is not divisible by 3,4, and 11 without remainder).



Start (First Scene)

This scene contains the title and an introduction to the program.



This scene has the following objects:

- Background
- Game instruction - add your own
- Group: buttons
 - button play
 - button introduction
 - button back

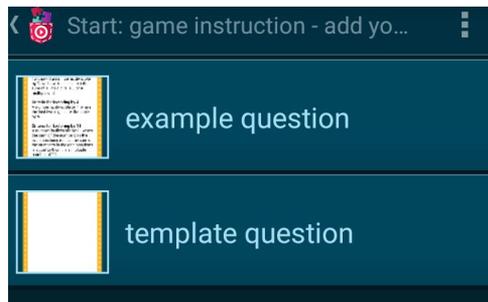


The object *Background* has 3 looks *title*, *instruction* and *game* which can be changed and adapted. The *game instruction* object shows the text that explains the objective of the game and the buttons *play*, *instruction* and *back* allow the player to read the instruction or start to play.

ToDo -1:

Create you own instruction

1. Open *game instruction* and go to Looks



2. Open the look *template question*: add the instruction of your game. You can achieve this in two ways:
 - a. Add the text by using Pocket Paint's text tool
 - b. Delete this image and add your own which contains your text

The group *buttons* contains 3 objects:

- Object *button play*: Starts the game – the game jumps to scene 2 *Action shooter example*
- Object *button instruction*: Tap this button to get to the instruction screen
- Object *button back*: Tap this button to go back to the title screen

Scene Action Shooter example level

In this scene you shoot comets with numbers on them. You have to shoot the numbers which can be divided by 3, 4, or 11 without remainder.

This scene contains the following objects:

- Background
- HUD elements
Elements that show a timer and the current score

- Group: game elements
 - correct shot
you gain a point by shooting this comet
 - wrong shot
you lose a point by shooting this comet
 - laser beam
laser for shooting comets
 - spaceship
main character of the game
- Display field
It is displayed at the end of the scene

ToDo - 2:

Play the game

1. Shoot a laser beam by tapping the spaceship



2. The goal is to shoot as many comets with the numbers that can be divided by 3, 4, and 11 on them as possible.
3. Avoid shooting the comets with other numbers

Learning goals:

- Revise the rules for division of a number by 3, 4, and 11 without remainder
- Compute the factorization of a number without paper or calculator

Scene Action Shooter template level

Now you also have to add some content and bricks to the program.

This scene contains the same objects as the scene **Action Shooter example**. In this case the objects *correct shot* and *wrong shot* are comets without numbers written on them.

The objective of this template is to define the new instructions of the game and add other conditions that make the content on comets correct or wrong. Following the topic of numbers factorization

there are more rules that we can use in order to create another game level, for instance:

- **Factoring by 5:** a number can be divided by 5 if the last digit is 5 or 0
- **Factoring by 7:** a number with digits **abc** can be divided by 7 if **ab-2c** is 0 or can be divided by 7. For example, 392 => $39 - (2 \times 2) = 35$ that is multiple of 7
- **Factoring by 13:** a number with digits **abc** can be divided by 13 if **ab-9c** is 0 or can be divided by 13. For example, 104 => $10 - (9 \times 4) = -26$ that is multiple of 13

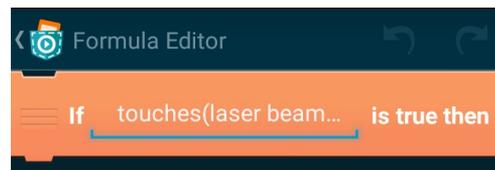
Edit the scene *Shooter template level*

ToDo - 3:

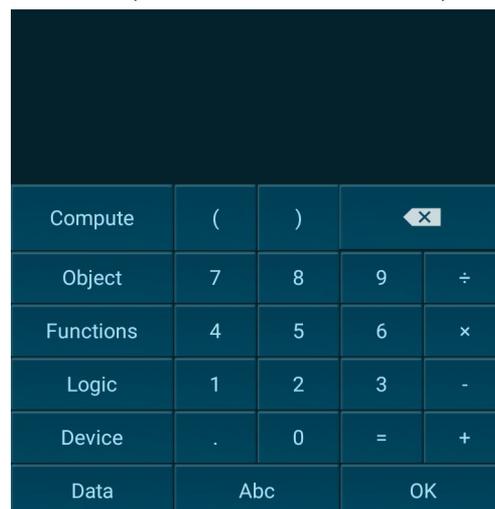
Add images and collisions for the correct and wrong shots

1. Open the looks of the *wrong shot* object and add a set of images with numbers that cannot be divided by 5, 7, and 13
2. Open the scripts of the *wrong shot* and follow the comments in the note bricks to subtract a point from the score if a collision with the object *laser beam* is detected
3. Open the looks of the *correct shot* object and add a set of images with numbers that can be divided by 5, 7, and 13
4. Open the scripts of the *correct shoot* and follow the comments in the note bricks to increase the score by one if a collision with the object *laser beam* is detected

Hint: try to practice a little with the rules and be sure that the numbers you chose are reasonable, the player should be able to do the mental arithmetic. For collision detection use **Formula Editor -> Object -> touches**



touches(laser beam - add code)



extra task - show level up

In this scene you should practice Create@School scripting. In order to do that the game elements group of the scene has an object called *extra task*



ToDo - 4:

Update the level without restarting the game. The objective of this extra task is to create a slow motion animation and then update the game level.

1. To accomplish this open the scripts of the *extra task* object and follow the instructions in the note bricks.



Scene End

This scene contains the following objects:

- Background
- Button Play Again

- Score display



Object *Button Play Again* starts the first scene again. *Score display* shows the total number of points collected by the player.

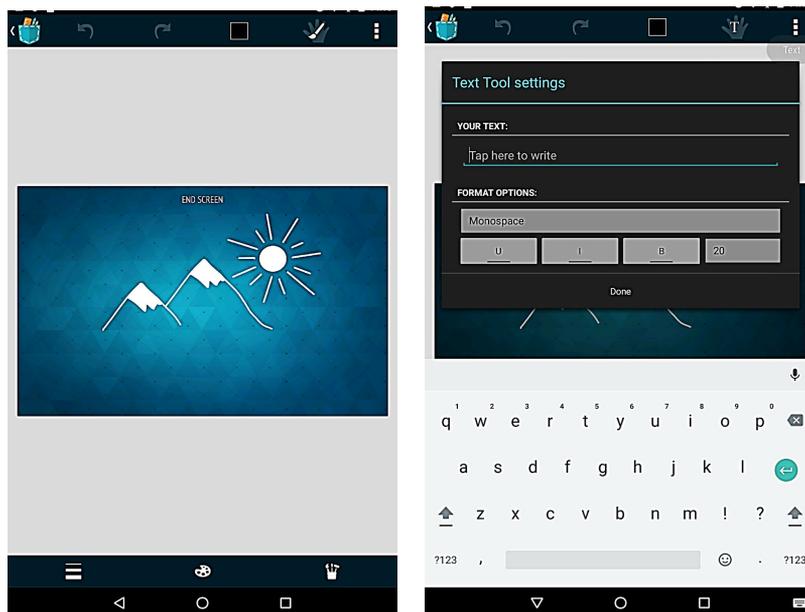
Edit the scene *End*

ToDo – 5:

Edit the End Screen

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

Pocket Paint:



Additionally, you can also add pictures:

3. Tools (bottom right) – Import image

