



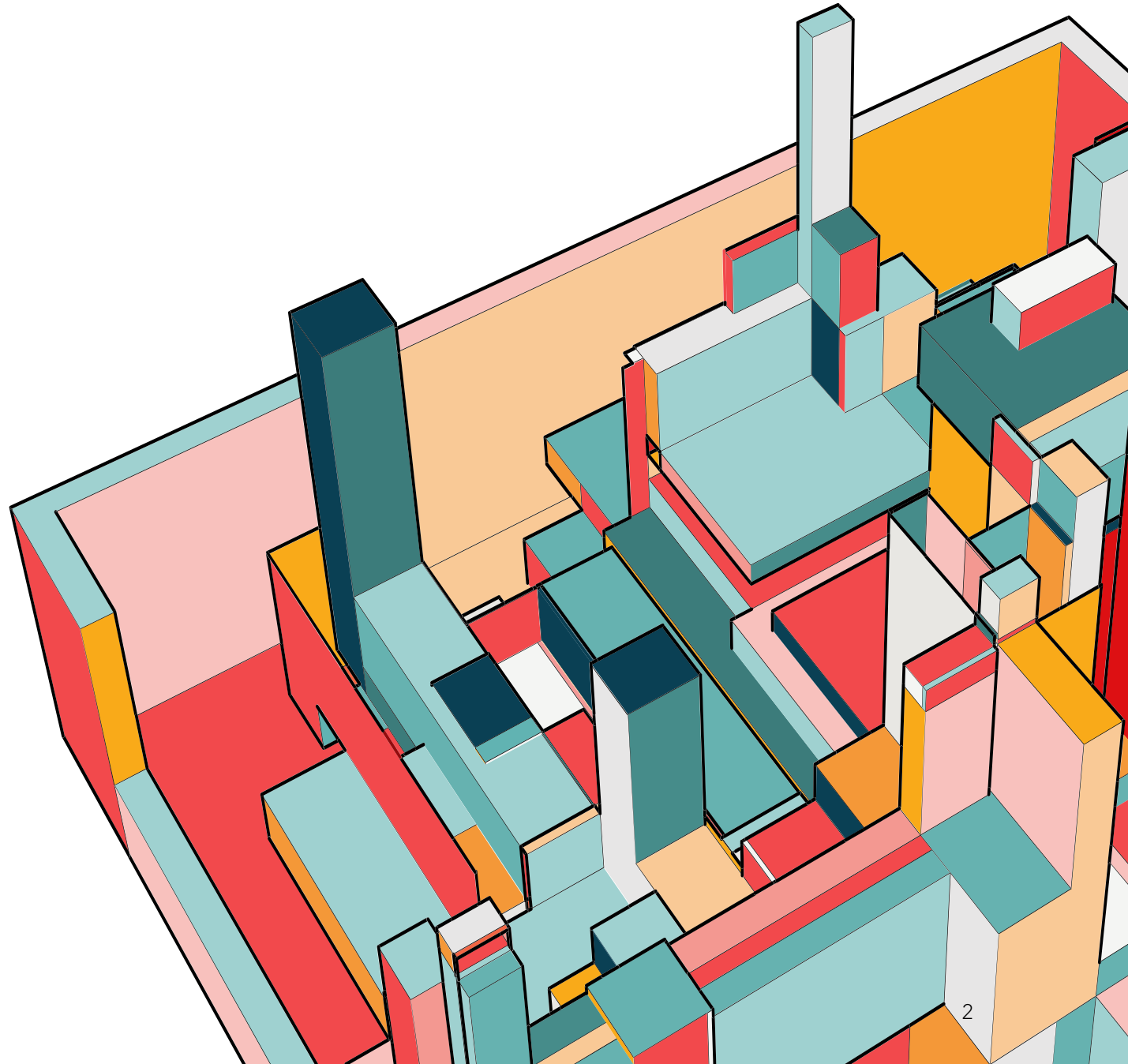
SPIELEENTWICKLUNG OHNE VORWISSEN

Session 3

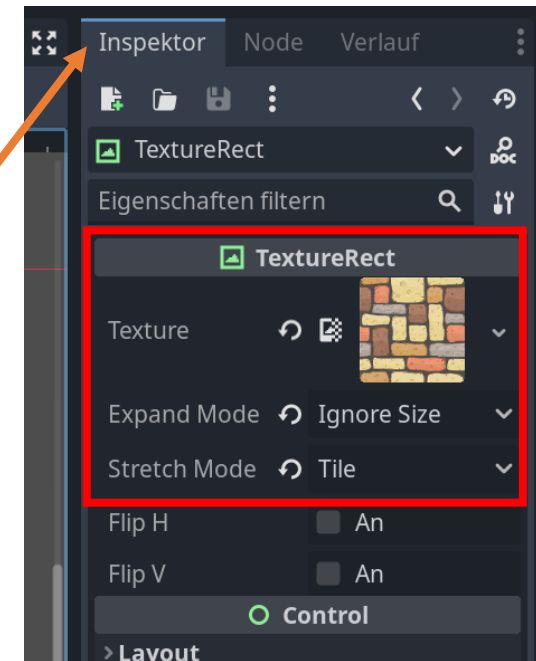
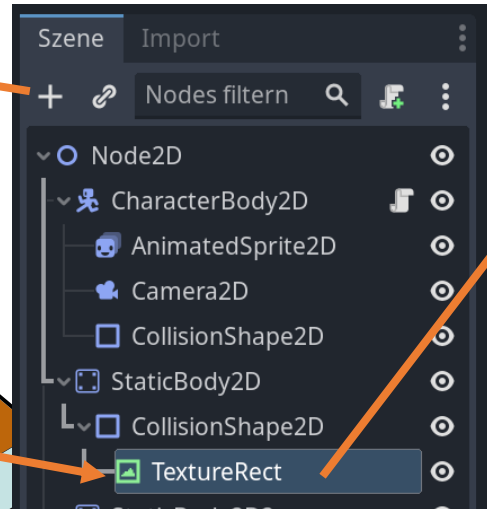
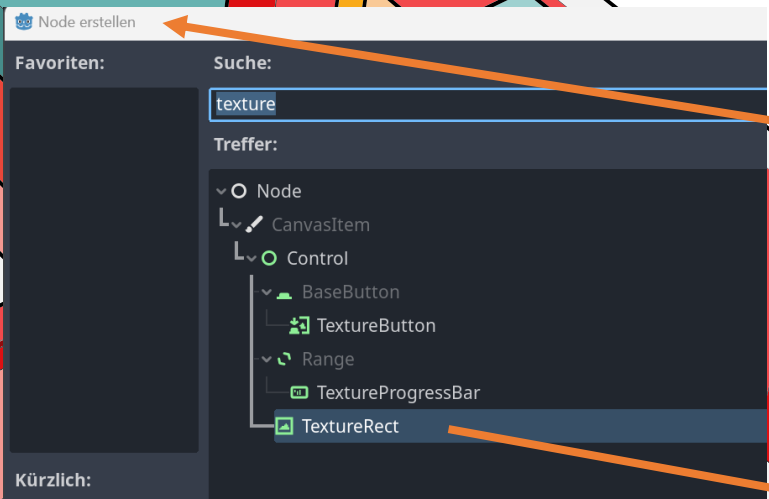
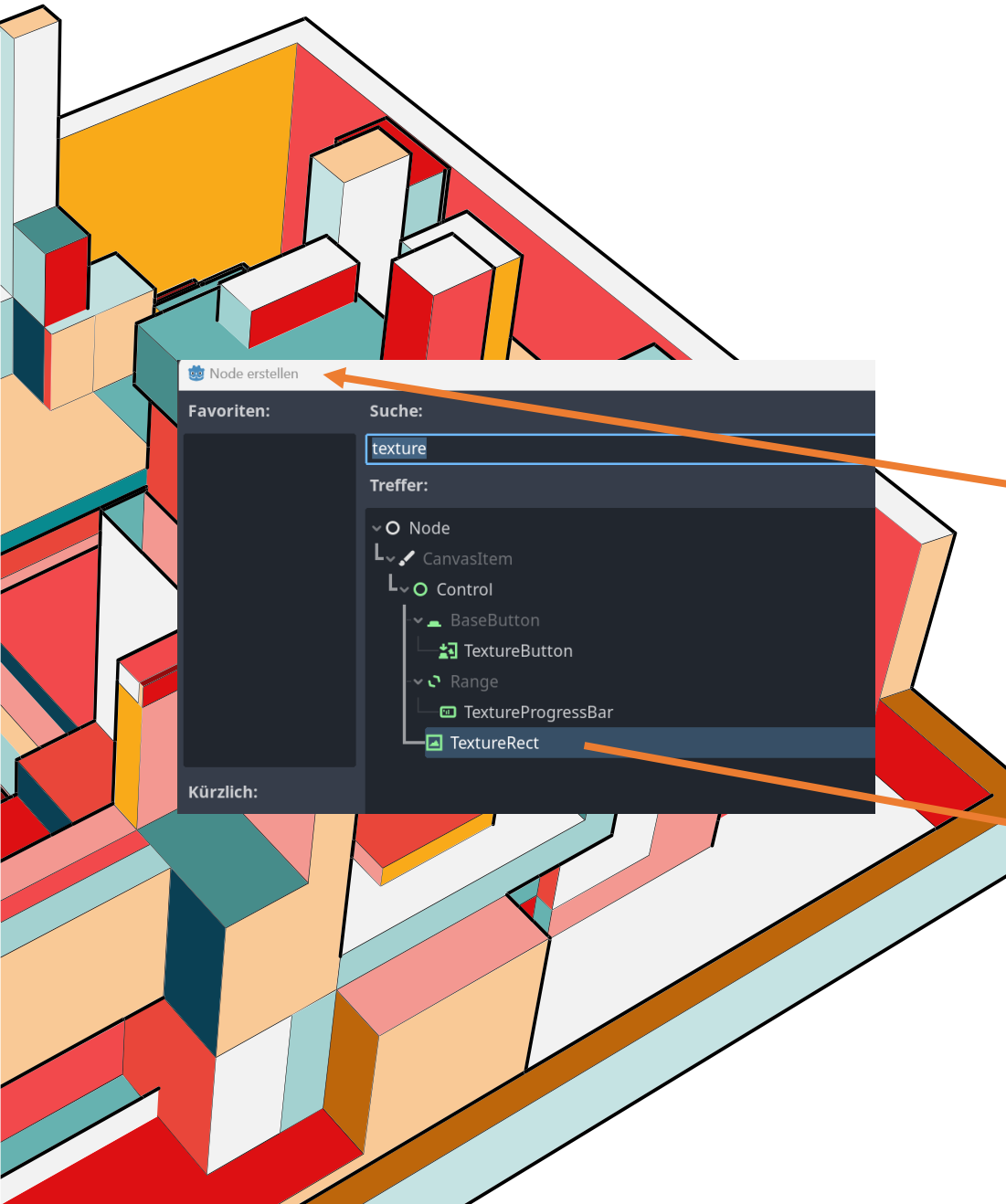
Hinweis:
Die Präsentationsfolien zeigen stets den einfachsten Weg, der
keine Programmierkenntnisse erfordert.

ROADMAP

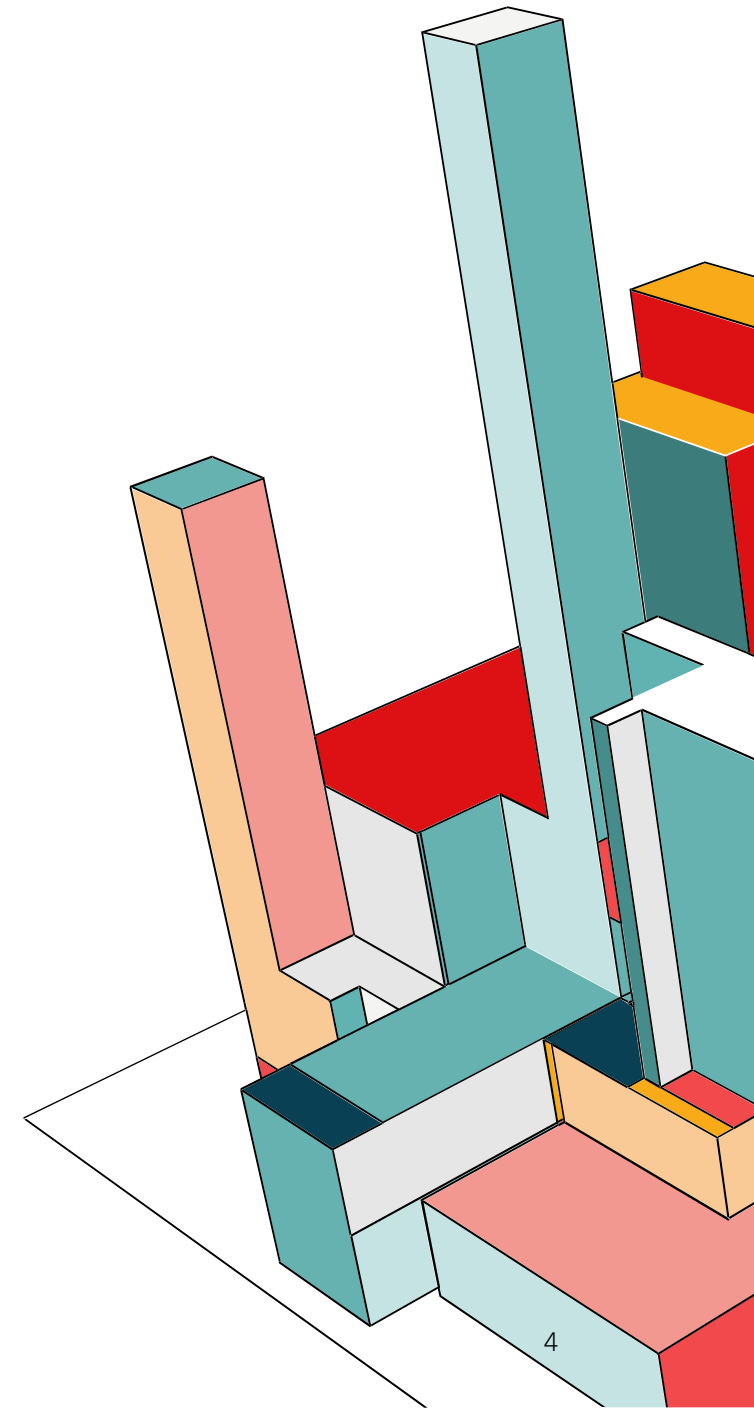
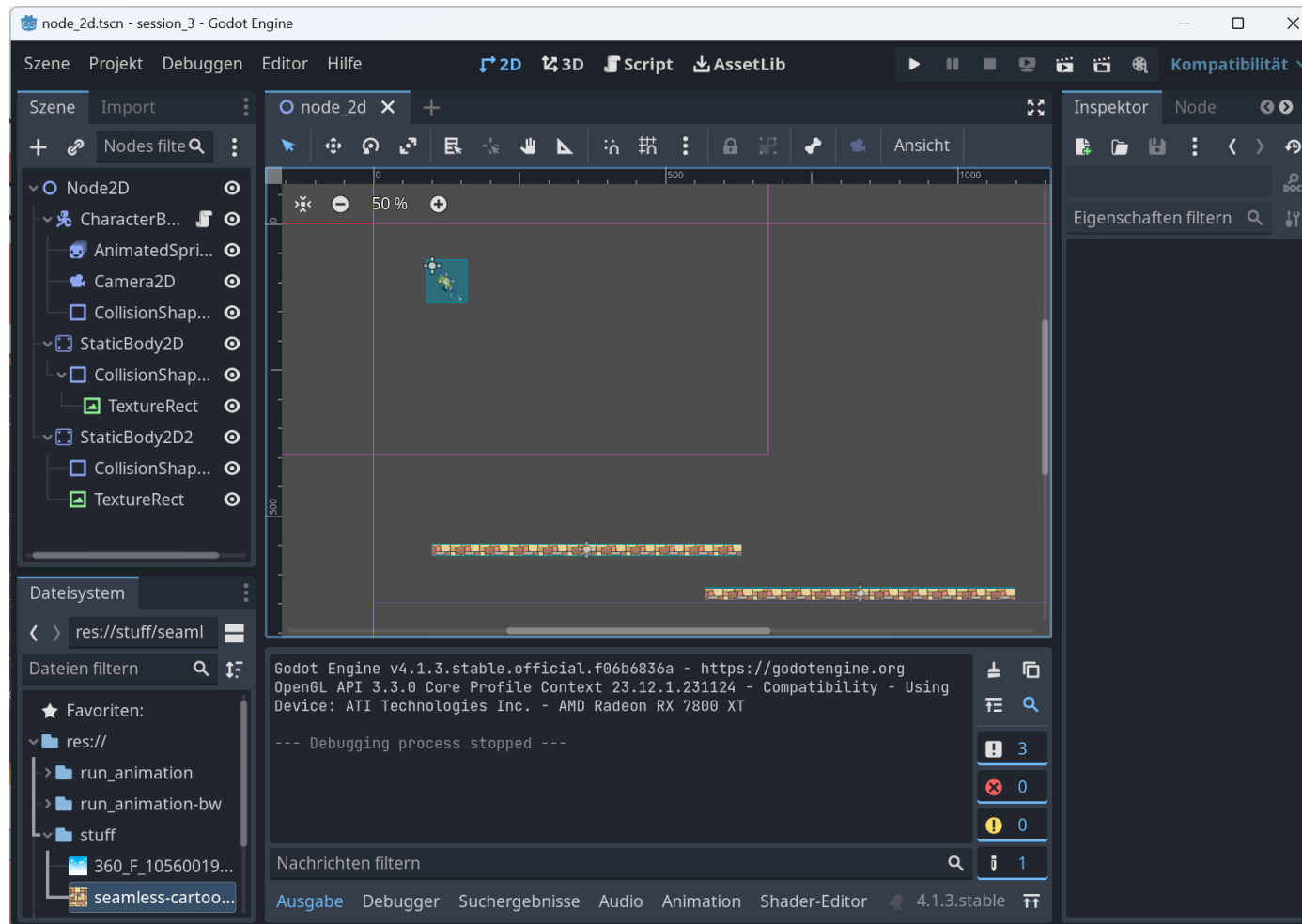
1. Texturen
2. Mechaniken



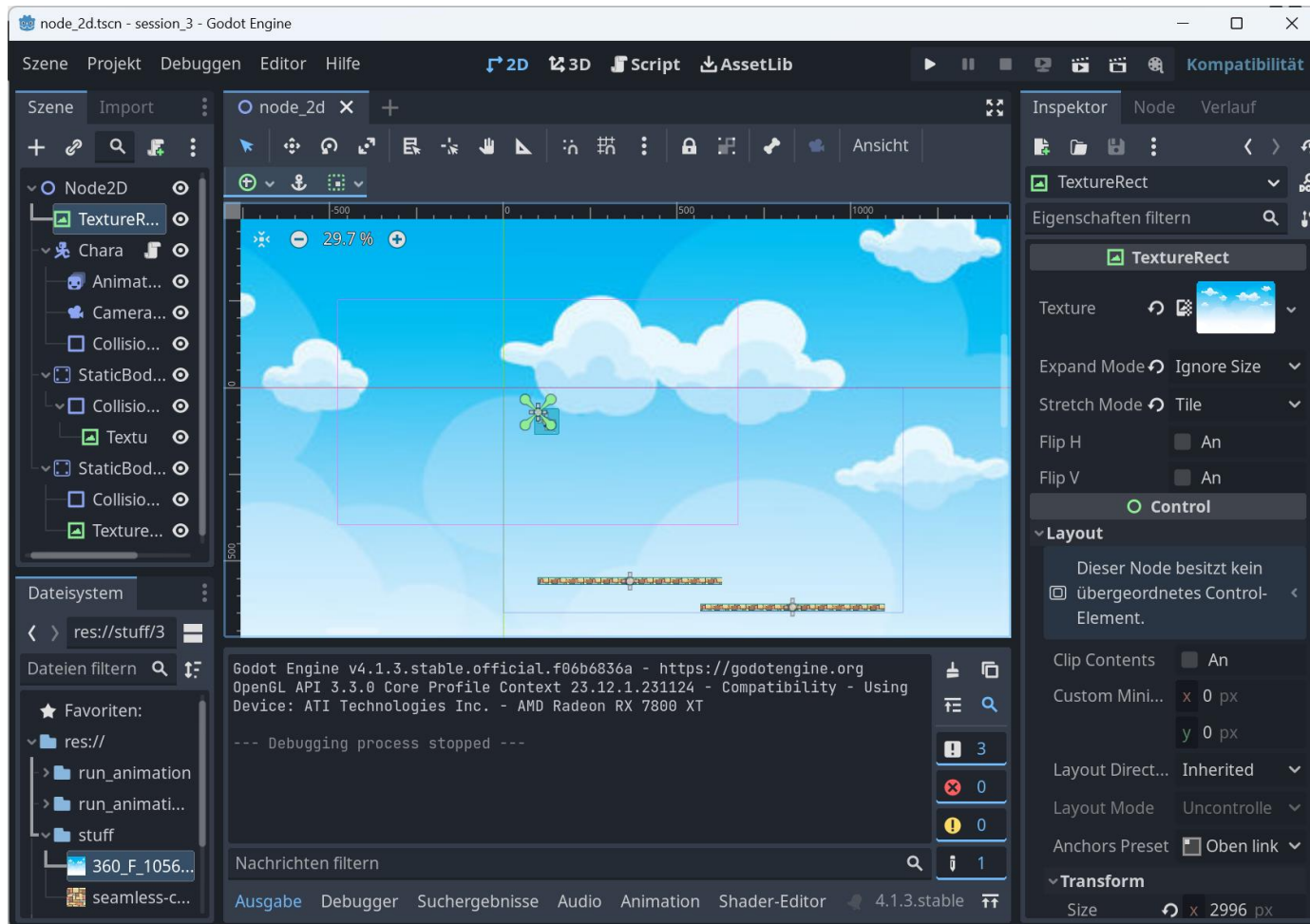
TEXTUREN

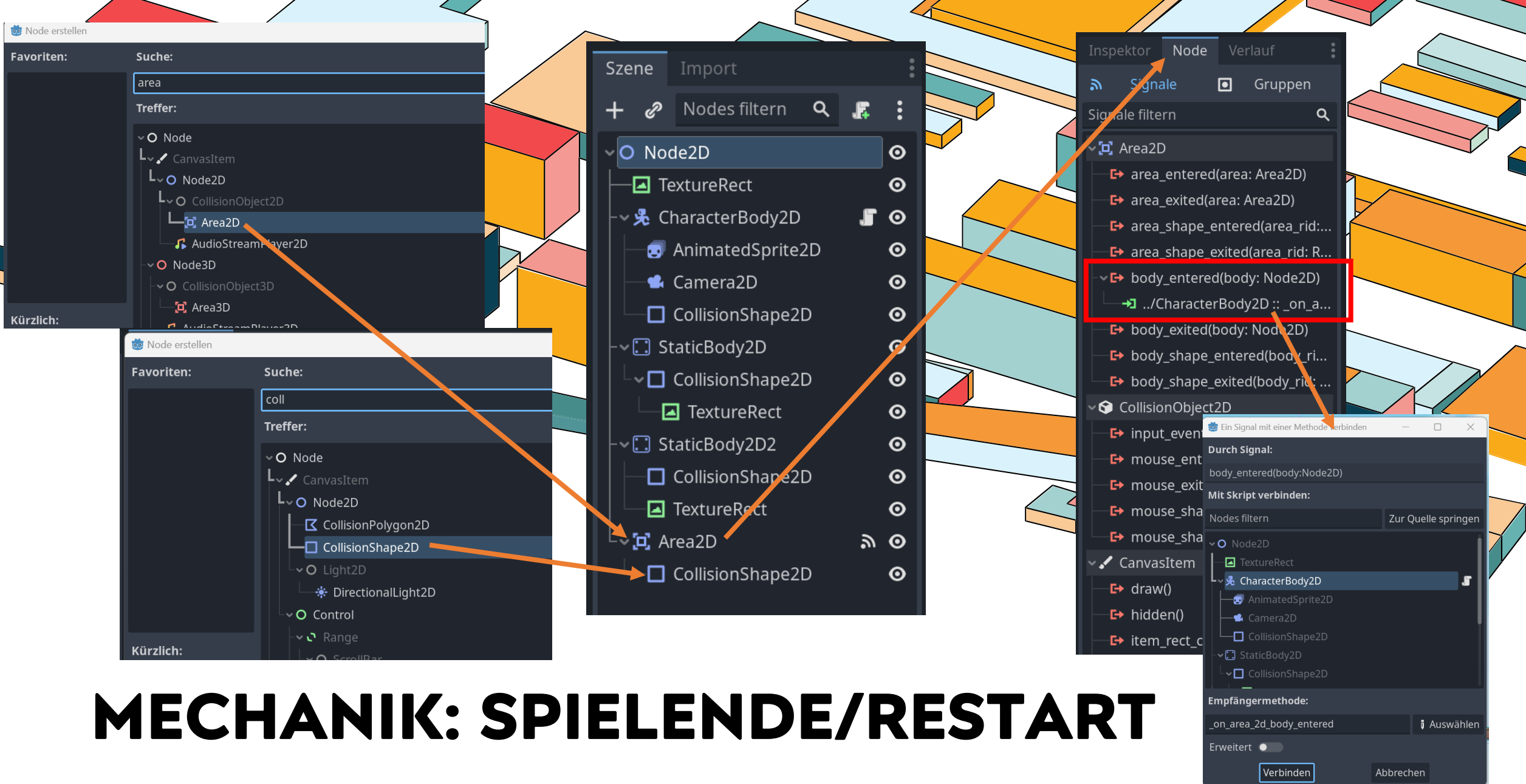


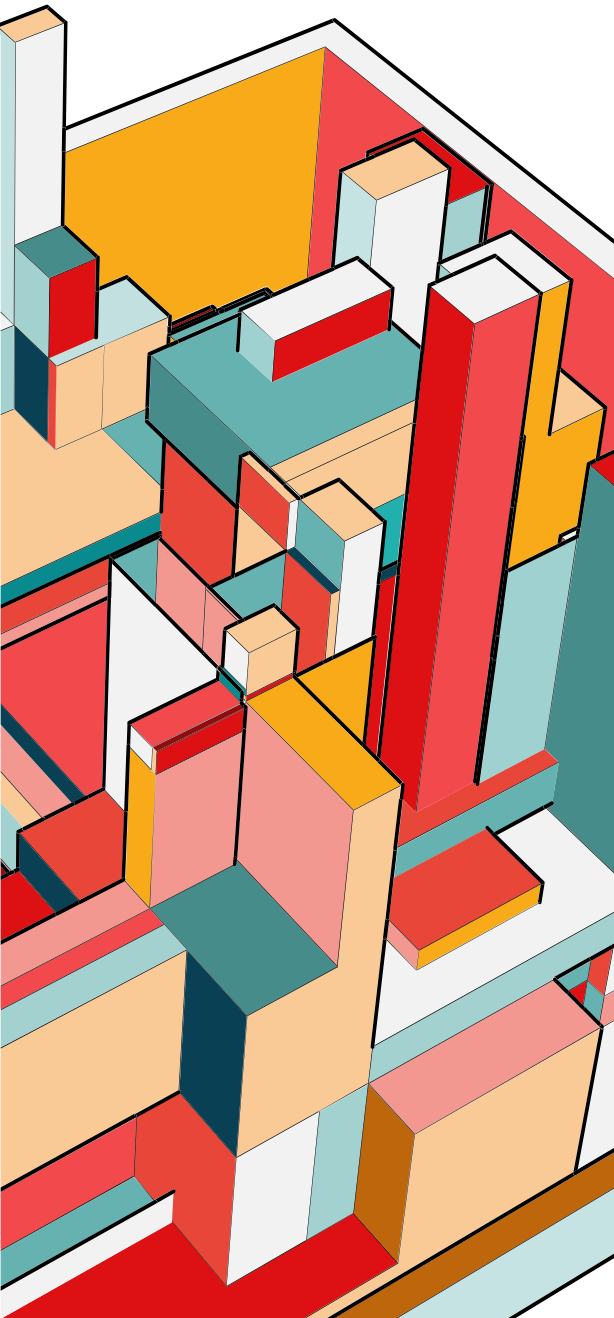
PLATTFORMEN ANPASSEN



HINTERGRUNDTEXTUR HINZUFÜGEN







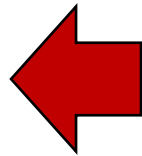
```
node_2d x +
Datei Bearbeiten Suchen Springe zu Debuggen

Skripte filtern
CharacterBody...
CollisionShape2...
AnimatedSprite...
Input

CharacterBody2D
Methoden filtern
_physics_process
_on_area_2d_body...

1 extends CharacterBody2D
2
3 @onready var _animated_sprite = $AnimatedSprite2D
4
5 const SPEED = 300.0
6 const JUMP_VELOCITY = -400.0
7
8 # Get the gravity from the project settings to be synced with RigidBody nodes.
9 var gravity = ProjectSettings.get_setting("physics/2d/default_gravity")
10
11 func _physics_process(delta):
12     # Add the gravity.
13     if not is_on_floor():
14         velocity.y += gravity * delta
15
16     # Handle Jump.
17     if Input.is_action_just_pressed("ui_accept") and is_on_floor():
18         velocity.y = JUMP_VELOCITY
19
20
21     if Input.is_action_pressed("ui_right"):
22         _animated_sprite.play("run_forward")
23     else:
24         if Input.is_action_pressed("ui_left"):
25             _animated_sprite.play("run_backwards")
26         else:
27             _animated_sprite.stop()
28
29     # Get the input direction and handle the movement/deceleration.
30     # As good practice, you should replace UI actions with custom gameplay actions.
31     var direction = Input.get_axis("ui_left", "ui_right")
32     if direction:
33         velocity.x = direction * SPEED
34     else:
35         velocity.x = move_toward(velocity.x, 0, SPEED)
36
37     move_and_slide()
38
39
40
41
42 func _on_area_2d_body_entered(body):
43     body.global_position = Vector2(0, 0)
44
```

VIELEN DANK



Folien und
Beispielcode
hier

