

Inhalt

Vorwort	XXIII
1 Einleitung	1
1.1 Multiplattform-Publishing	1
1.2 Das kann Unity (nicht)	2
1.3 Lizenzmodelle	2
1.4 Aufbau und Ziel des Buches	3
1.5 Weiterentwicklung von Unity	4
1.6 Online-Zusatzmaterial	4
2 Grundlagen	7
2.1 Installation	7
2.2 Oberfläche	7
2.2.1 Hauptmenü	9
2.2.2 Scene View	10
2.2.2.1 Navigieren in der Scene View	11
2.2.2.2 Scene Gizmo	11
2.2.3 Game View	12
2.2.4 Toolbar	13
2.2.5 Hierarchy	15
2.2.5.1 Parenting	16
2.2.6 Inspector	17
2.2.6.1 Kopfinformationen im Inspector	18
2.2.6.2 Variablen Werte zuweisen	18
2.2.6.3 Komponenten-Menüs	19
2.2.6.4 Preview-Fenster	20
2.2.7 Project Brower	20
2.2.7.1 Assets suchen und finden	21
2.2.7.2 Assets importieren	22
2.2.8 Console	22

2.3	Das Unity-Projekt	23
2.3.1	Neues Projekt anlegen	23
2.3.2	Bestehendes Projekt öffnen	24
2.3.2.1	Ältere Projekte öffnen	25
2.3.3	Projektdateien	25
2.3.4	Szene	25
2.3.5	Game Objects	26
2.3.6	Components	27
2.3.7	Tags	28
2.3.8	Layer	29
2.3.9	Assets	29
2.3.9.1	UnityPackage	30
2.3.9.2	Asset Import	30
2.3.9.3	Asset Export	31
2.3.9.4	Asset Store	31
2.3.10	Frames	32
2.4	Das erste Übungsprojekt	33
3	C# und Unity	35
3.1	Die Sprache C#	35
3.2	Syntax	36
3.3	Kommentare	37
3.4	Variablen	37
3.4.1	Namenskonventionen	37
3.4.2	Datentypen	38
3.4.3	Schlüsselwort var	39
3.4.4	Datenfelder/Array	39
3.4.4.1	Arrays erstellen	39
3.4.4.2	Zugriff auf ein Array-Element	40
3.4.4.3	Anzahl aller Array-Items ermitteln	40
3.4.4.4	Mehrdimensionale Arrays	40
3.5	Konstanten	41
3.5.1	Enumeration	41
3.6	Typkonvertierung	42
3.7	Rechnen	43
3.8	Verzweigungen	44
3.8.1	if-Anweisungen	44
3.8.1.1	Komplexere if-Anweisungen	45
3.8.2	switch-Anweisung	46
3.9	Schleifen	47
3.9.1	for-Schleife	47
3.9.1.1	Negative Schrittweite	48
3.9.1.2	break	48

3.9.2 Foreach-Schleife	48
3.9.3 while-Schleife	49
3.9.4 do-Schleife	49
3.10 Klassen	50
3.10.1 Komponenten per Code zuweisen	50
3.10.2 Instanziierung von Nichtkomponenten	51
3.10.3 Werttypen und Referenztypen	53
3.10.4 Überladene Methoden	53
3.11 Der Konstruktor	54
3.11.1 Konstruktoren in Unity	54
3.12 Lokale und globale Variablen	55
3.12.1 Namensverwechslung verhindern mit this	55
3.13 Zugriff und Sichtbarkeit	55
3.14 Statische Klassen und Klassenmember	56
3.15 Parametermodifizierer out/ref	57
3.16 Array-Übergabe mit params	58
3.17 Eigenschaften und Eigenschaftsmethoden	59
3.18 Vererbung	60
3.18.1 Basisklasse und abgeleitete Klassen	60
3.18.2 Vererbung und die Sichtbarkeit	61
3.18.3 Geerbte Methode überschreiben	61
3.18.4 Zugriff auf die Basisklasse	62
3.18.5 Klassen versiegeln	62
3.19 Polymorphie	63
3.20 Schnittstellen	63
3.20.1 Schnittstelle definieren	63
3.20.2 Schnittstellen implementieren	64
3.20.2.1 Unterstützung durch MonoDevelop	65
3.20.3 Zugriff über eine Schnittstellen	65
3.21 Namespaces	66
3.21.1 Eigene Namespaces definieren	66
3.22 Generische Klassen und Methoden	67
3.22.1 List	68
3.22.2 Dictionary	69
4 Skript-Programmierung	71
4.1 MonoDevelop	71
4.1.1 Hilfe in MonoDevelop	72
4.1.2 Syntaxfehler	72
4.2 Nutzbare Programmiersprachen	73
4.2.1 Warum C#?	73
4.3 Unitys Vererbungsstruktur	74
4.3.1 Object	74

4.3.2	GameObject	75
4.3.3	ScriptableObject	75
4.3.4	Component	75
4.3.5	Transform	76
4.3.6	Behaviour	76
4.3.7	MonoBehaviour	76
4.4	Skripte erstellen	76
4.4.1	Skripte umbenennen	77
4.5	Das Skript-Grundgerüst	77
4.6	Unitys Event-Methoden	78
4.6.1	Update	78
4.6.2	FixedUpdate	79
4.6.3	Awake	80
4.6.4	Start	80
4.6.5	OnGUI	80
4.6.6	LateUpdate	81
4.7	Komponentenprogrammierung	81
4.7.1	Auf GameObjects zugreifen	82
4.7.1.1	FindWithTag	82
4.7.1.2	FindGameObjectsWithTag	83
4.7.1.3	Find	83
4.7.2	GameObjects aktivieren und deaktivieren	83
4.7.3	GameObjects zerstören	84
4.7.4	GameObjects erstellen	84
4.7.5	Auf Components zugreifen	84
4.7.5.1	GetComponent	85
4.7.5.2	SendMessage	85
4.7.5.3	Direktzugriff	86
4.7.6	Components hinzufügen	86
4.7.7	Components entfernen	87
4.7.8	Components aktivieren und deaktivieren	87
4.7.9	Attribute	87
4.8	Zufallswerte	89
4.9	Parallel Code ausführen	90
4.9.1	WaitForSeconds	91
4.10	Verzögerte und wiederholende Funktionsaufrufe mit Invoke	91
4.10.1	Invoke	91
4.10.2	InvokeRepeating, IsInvoking und CancelInvoke	92
4.11	Daten speichern und laden	93
4.11.1	PlayerPrefs-Voreinstellungen	93
4.11.2	Daten speichern	94
4.11.3	Daten laden	94
4.11.4	Key überprüfen	95

4.11.5 Löschen	95
4.11.6 Save	95
4.12 Szeneübergreifende Daten	96
4.12.1 Werteübergabe mit PlayerPrefs	96
4.12.1.1 Startmenüs zur Initialisierung nutzen	97
4.12.2 Zerstörung unterbinden	98
4.12.2.1 DontDestroyOnLoad als Singleton	98
4.13 Debug-Klasse	99
4.14 Kompilierungsreihenfolge	100
4.14.1 Programmiersprachen mischen und der sprachübergreifende Zugriff	100
4.15 Ausführungsreihenfolge	101
4.16 Plattformabhängig Code kompilieren	101
4.17 Eigene Assets mit ScriptableObject	102
4.17.1 Neue ScriptableObject-Subklasse erstellen	103
4.17.2 Instanzen eines ScriptableObjects erstellen	104
4.17.2.1 Eigener Menü-Knopf zum Erstellen von Assets	104
5 Objekte in der zweiten und dritten Dimension	107
5.1 Das 3D-Koordinatensystem	107
5.2 Vektoren	108
5.2.1 Ort, Winkel und Länge	109
5.2.2 Normalisieren	110
5.3 Das Mesh	111
5.3.1 Normalenvektor	112
5.3.2 MeshFilter und MeshRenderer	112
5.4 Transform	114
5.4.1 Kontextmenü der Transform-Komponente	114
5.4.2 Objekthierarchien	115
5.4.3 Scripting mit Transform	116
5.4.4 Quaternion	116
5.5 Shader und Materials	117
5.5.1 Der Standard-Shader	118
5.5.1.1 Rendering Mode	120
5.5.1.2 Main Maps	121
5.5.1.3 Secondary Maps	127
5.5.1.4 Metallic vs. Specular	129
5.5.2 Texturen	133
5.5.2.1 Import von Texturen	134
5.5.2.2 Normalmaps erstellen	135
5.5.3 UV Mapping	136
5.6 3D-Modelle einer Szene zufügen	138
5.6.1 Primitives	138

5.6.2	3D-Modelle importieren	139
5.6.2.1	Model Import Settings	140
5.6.3	In Unity modellieren	140
5.6.4	Prozedurale Mesh-Generierung	141
5.6.5	Level Of Detail	141
5.6.5.1	LODGroups	142
5.6.5.2	LOD-Qualitätseinstellungen	143
5.7	2D in Unity	143
5.7.1	Sprites	144
5.7.1.1	Sprite Editor	146
5.7.1.2	Sprite Packer	147
5.7.2	SpriteRenderer	149
5.7.2.1	Darstellungsreihenfolgen von Sprites	150
5.7.2.2	Sprite-Shader	151
5.7.3	Parallax Scrolling	152
6	Kameras, die Augen des Spielers	155
6.1	Die Kamera	155
6.1.1	Komponenten eines Kamera-Objektes	157
6.1.2	HDR - High Dynamic Range-Rendering	157
6.1.2.1	Rückwandeln mit Tonemapping	158
6.1.2.2	Abarbeitungsreihenfolge festlegen	159
6.1.3	Linearer- und Gamma-Farbraum	160
6.1.3.1	Die Gamma-Korrektur	160
6.1.3.2	Linear-Space und Gamma-Space in Unity	161
6.2	Kamerasteuerung	163
6.2.1	Statische Kamera	163
6.2.2	Parenting-Kamera	163
6.2.3	Kamera-Skripte	164
6.2.3.1	Kamera-Skripte programmieren	164
6.3	ScreenPointToRay	166
6.4	Mehrere Kameras	166
6.4.1	Kamerawechsel	166
6.4.2	Split-Screen	168
6.4.3	Einfache Minimap	169
6.4.4	Render Texture	170
6.5	Image Effects	172
6.5.1	Beispiel: Haus bei Nacht	172
6.6	Skybox	174
6.6.1	Mehrere Skyboxen gleichzeitig einsetzen	175
6.6.2	Skybox selber erstellen	176
6.7	Occlusion Culling	177
6.7.1	Occluder Static und Occludee Static	178
6.7.2	Occlusion Culling erstellen	179

7	Licht und Schatten	181
7.1	Ambient Light	181
7.2	Lichtarten	182
7.2.1	Directional Light	183
7.2.2	Point Light	184
7.2.3	Spot Light	184
7.2.4	Area Light	185
7.3	Schatten	186
7.3.1	Einfluss des MeshRenderers auf Schatten	187
7.4	Light Cookies	188
7.4.1	Import Settings eines Light Cookies	188
7.4.2	Light Cookies und Point Lights	189
7.5	Light Halos	190
7.5.1	Unabhängige Halos	191
7.6	Lens Flares	191
7.6.1	Eigene Lens Flares	192
7.7	Projector	192
7.7.1	Standard Projectors	192
7.8	Lightmapping	194
7.8.1	Light Probes	196
7.9	Rendering Paths	198
7.9.1	Forward Rendering	199
7.9.2	Vertex Lit	200
7.9.3	Deferred Lighting	201
7.10	Global Illumination	202
7.10.1	Baked GI	202
7.10.2	Precomputed Realtime GI	203
7.11	Reflexionen (Spiegelungen)	203
7.11.1	Reflection Probes	204
7.12	Qualitätseinstellungen	207
7.12.1	Quality Settings	207
7.12.2	Qualitätsstufen per Code festlegen	207
8	Physik in Unity	209
8.1	Physikberechnung	209
8.2	Rigidbodies	210
8.2.1	Rigidbodies kennenlernen	211
8.2.2	Masseschwerpunkt	212
8.2.3	Kräfte und Drehmomente zufügen	213
8.2.3.1	AddForce-Methode	213
8.2.3.2	AddTorque-Methode	214
8.2.3.3	ConstantForce-Komponente	216

8.3	Kollisionen	216
8.3.1	Collider	216
8.3.1.1	Mesh Collider	217
8.3.1.2	Collider modifizieren	218
8.3.1.3	OnCollision-Methoden	219
8.3.2	Trigger	220
8.3.3	Static Collider	221
8.3.4	Kollisionen mit schnellen Objekten	221
8.3.5	Terrain Collider	223
8.3.6	Layer-basierende Kollisionserkennung	223
8.3.7	Mit Layer-Masken arbeiten	223
8.4	Wheel Collider	225
8.4.1	Wheel Friction Curve	226
8.4.2	Entwicklung einer Fahrzeugsteuerung	228
8.4.2.1	CarController.cs	233
8.4.3	Autokonfiguration	235
8.4.4	Fahrzeugstabilität	237
8.5	Physic Materials	238
8.6	Joints	239
8.6.1	Fixed Joint	239
8.6.2	Spring Joint	239
8.6.3	Hinge Joint	239
8.7	Raycasting	240
8.8	Character Controller	241
8.8.1	SimpleMove	242
8.8.2	Move	243
8.8.3	Kräfte zufügen	244
8.8.4	Einfacher First Person Controller	245
8.9	2D-Physik	247
8.9.1	OnCollision2D- und OnTrigger2D-Methoden	248
8.9.2	2D Physic Effectors	250
9	Maus, Tastatur, Touch	251
9.1	Virtuelle Achsen und Tasten	251
9.1.1	Der Input-Manager	251
9.1.2	Virtuelle Achsen	253
9.1.3	Virtuelle Tasten	253
9.1.4	Steuern mit Mauseingaben	254
9.1.5	Joystick-Inputs	254
9.1.6	Anlegen neuer Inputs	255
9.2	Achsen- und Tasteneingaben auswerten	255
9.2.1	GetAxis	255
9.2.2	GetButton	256

9.3	Tastatureingaben auswerten	257
9.3.1	GetKey	257
9.3.2	anyKey	257
9.4	Mauseingaben auswerten	258
9.4.1	GetMouseButton	258
9.4.2	mousePosition	259
9.4.3	Mauszeiger ändern	260
9.5	Touch-Eingaben auswerten	261
9.5.1	Der Touch-Typ	261
9.5.2	Input.touches	262
9.5.3	TouchCount	262
9.5.4	GetTouch	263
9.5.5	CrossPlatformInput	263
9.6	Beschleunigungssensor auswerten	265
9.6.1	Input.acceleration	265
9.6.2	Tiefpass-Filter	266
9.7	Steuerungen bei Mehrspieler-Games	267
9.7.1	Split-Screen-Steuerung	267
9.7.2	Netzwerkspiele	268
10	Audio	269
10.1	AudioListener	269
10.2	AudioSource	270
10.2.1	Durch Mauern hören verhindern	272
10.2.2	Sound starten und stoppen	274
10.2.3	Temporäre AudioSource	275
10.3	AudioClip	276
10.3.1	Länge ermitteln	276
10.4	Reverb Zone	276
10.5	Filter	278
10.6	Audio Mixer	278
10.6.1	Das Audio Mixer-Fenster	278
10.6.1.1	Die Audio-Steuerungseinheit	280
10.6.2	Audiosignalwege	282
10.6.2.1	Send & Receive	283
10.6.2.2	Duck Volume	284
10.6.3	Mit Snapshots arbeiten	286
10.6.4	Views erstellen	287
11	Partikeleffekte mit Shuriken	289
11.1	Editor-Fenster	290
11.2	Particle Effect Control	291
11.3	Numerische Parametervarianten	291

11.4	Farbparameter-Varianten	292
11.5	Default-Modul	292
11.6	Effekt-Module	293
11.6.1	Emission	294
11.6.2	Shape	294
11.6.2.1	Sphere	294
11.6.2.2	HemiSphere	294
11.6.2.3	Cone	294
11.6.2.4	Box	295
11.6.2.5	Mesh	295
11.6.3	Velocity over Lifetime	295
11.6.4	Limit Velocity over Lifetime	296
11.6.5	Force over Lifetime	296
11.6.6	Color over Lifetime	296
11.6.7	Color by Speed	296
11.6.8	Size over Lifetime	297
11.6.9	Size by Speed	297
11.6.10	Rotation over Lifetime	297
11.6.11	Rotation by Speed	297
11.6.12	External Forces	297
11.6.13	Collision	298
11.6.13.1	Planes	298
11.6.13.2	World	299
11.6.14	Sub Emitter	299
11.6.15	Texture-Sheet-Animation	299
11.6.16	Renderer	300
11.6.16.1	Billboard	301
11.6.16.2	Stretched Billboard	301
11.6.16.3	Horizontal Billboard	301
11.6.16.4	Vertical Billboard	301
11.6.16.5	Mesh	302
11.7	Partikelemission starten, stoppen und unterbrechen	302
11.7.1	Play	302
11.7.2	Stop	303
11.7.3	Pause	303
11.7.4	enableEmission	303
11.8	OnParticleCollision	303
11.8.1	GetCollisionEvents	304
11.9	Feuer erstellen	304
11.9.1	Materials erstellen	305
11.9.2	Feuer-Partikelsystem	306
11.9.3	Rauch-Partikelsystem	309
11.10	Wassertropfen erstellen	312
11.10.1	Tropfen-Material erstellen	312

11.10.2	Wassertropfen-Partikelsystem	313
11.10.3	Kollisionspartikelsystem	315
11.10.4	Kollisionssound	317
12	Landschaften gestalten	319
12.1	Was Terrains können und wo die Grenzen liegen	320
12.2	Terrainhöhe verändern	320
12.2.1	Pinsel	321
12.2.2	Oberflächen anheben und senken	321
12.2.3	Plateaus und Schluchten erstellen	322
12.2.4	Oberflächen weicher machen	323
12.2.5	Heightmaps	323
12.2.5.1	Reale Landschaften als Vorlagen nutzen	324
12.3	Terrain texturieren	325
12.3.1	Textur-Pinsel	326
12.3.2	Textures verwalten	326
12.4	Bäume und Sträucher	328
12.4.1	Bedienung des Place Tree-Tools	328
12.4.2	Wälder erstellen	329
12.4.3	Mit Bäumen kollidieren	329
12.5	Gräser und Details hinzufügen	330
12.5.1	Detail-Meshs	330
12.5.2	Gräser	331
12.5.3	Quelldaten nachladen	332
12.6	Terrain-Einstellungen	332
12.6.1	Base Terrain	333
12.6.2	Resolution	333
12.6.3	Tree & Details Objects	334
12.6.4	Wind Settings	334
12.6.5	Zur Laufzeit Terrain-Eigenschaften verändern	335
12.7	Der Weg zum perfekten Terrain	336
12.8	Gewässer	337
13	Wind Zones	339
13.1	Spherical vs. Directional	340
13.2	Wind Zone - Eigenschaften	341
13.3	Frische Brise	342
13.4	Turbine	342
14	GUI	343
14.1	Das UI-System uGUI	344
14.1.1	Canvas	344
14.1.1.1	Render Modes	345
14.1.1.2	Canvas Scaler	347

14.1.2	RectTransform	348
14.1.2.1	Anchors	350
14.1.3	UI-Sprite Import	352
14.1.4	Grafische Controls	353
14.1.4.1	Text	353
14.1.4.2	Image	354
14.1.4.3	Raw Image	355
14.1.4.4	Panel	355
14.1.4.5	Effects	355
14.1.5	Interaktive Controls	356
14.1.5.1	Button	359
14.1.5.2	InputField	360
14.1.5.3	Toggle und Toggle Group	360
14.1.5.4	Slider	361
14.1.5.5	Scrollbar	362
14.1.6	Controls designen	362
14.1.7	Animationen in uGUI	363
14.1.8	Event Trigger	364
14.2	Screen-Klasse	365
14.2.1	Schriftgröße dem Bildschirm anpassen	366
14.3	OnGUI-Programmierung	366
14.3.1	GUI	367
14.3.1.1	Label	367
14.3.1.2	Button und RepeatButton	368
14.3.1.3	TextField und TextArea	368
14.3.1.4	Subfenster	369
14.3.2	GUILayout	370
14.3.2.1	Ausrichtung festlegen	370
14.3.3	GUILayout und GUIStyle	371
15	Prefabs	373
15.1	Prefabs erstellen und nutzen	373
15.2	Prefab-Instanzen erzeugen	373
15.2.1	Instanzen per Code erstellen	374
15.2.2	Instanzen weiter bearbeiten	374
15.3	Prefabs ersetzen und zurücksetzen	375
16	Internet und Datenbanken	377
16.1	Die WWW-Klasse	377
16.1.1	Rückgabewert-Formate	378
16.1.2	Parameter übergeben	379
16.1.2.1	WWWForm	379
16.2	Datenbank-Kommunikation	380
16.2.1	Daten in einer Datenbank speichern	380

16.2.2	Daten von einer Datenbank abfragen	381
16.2.3	Rückgabewerte parsen	383
16.2.4	Datenhaltung in eigenen Datentypen	383
16.2.5	HighscoreCommunication.cs	385
16.2.6	Datenbankverbindung in PHP	387
17	Animationen	389
17.1	Allgemeiner Animation-Workflow	390
17.2	Animationen erstellen	390
17.2.1	Animation View	391
17.2.2	Curves vs. Dope Sheet	392
17.2.3	Animationsaufnahme	392
17.2.3.1	Ansicht im Animation-Fenster anpassen	393
17.2.3.2	Sprite-Animationen	393
17.2.3.3	Root Motion-Kurven erstellen	396
17.2.4	Beispiel Fallgatter-Animation	397
17.3	Animationen importieren	398
17.3.1	Rig	399
17.3.1.1	Generic	399
17.3.1.2	Humanoid	399
17.3.2	Animationen	401
17.3.2.1	Animationen aufteilen	401
17.3.2.2	Importierte Animation-Clips bearbeiten	402
17.3.2.3	Mask	404
17.3.2.4	Events	404
17.4	Animationen einbinden	404
17.4.1	Animator Controller	404
17.4.1.1	Animation States erstellen	405
17.4.1.2	Transitions – Animation States wechseln	406
17.4.1.3	Any State nutzen	408
17.4.1.4	Blend Trees	408
17.4.1.5	Layer und Avatar Masks	409
17.4.1.6	Sub-State Machines	411
17.4.1.7	State Machine Behaviours	416
17.4.2	Animator-Komponente	420
17.4.3	Beispiel Fallgatter: Animator Controller	421
17.5	Controller-Skripte	423
17.5.1	Parameter des Animator Controllers setzen	423
17.5.1.1	Mit Hash-Werten arbeiten	424
17.5.2	Animation States abfragen	424
17.5.3	Beispiel Fallgatter Controller-Skript	425
17.6	Animation Events	427

18	Künstliche Intelligenz	429
18.1	NavMeshAgent	430
18.1.1	Eigenschaften der Navigationskomponente	431
18.1.2	Zielpunkt zuweisen	431
18.1.3	Pfadsuche unterbrechen und fortsetzen	432
18.2	NavigationMesh	432
18.2.1	Object Tab	434
18.2.2	Bake Tab	434
18.2.3	Areas Tab	435
18.3	NavMeshObstacle	436
18.4	Off-Mesh Link	437
18.4.1	Automatische Off-Mesh Links	437
18.4.2	Manuelle Off-Mesh Links	438
18.5	Point & Click-Steuerung für Maus und Touch	439
19	Fehlersuche und Performance	443
19.1	Fehlersuche	443
19.1.1	Breakpoints	444
19.1.2	Variablen beobachten	445
19.1.3	Console Tab nutzen	445
19.1.4	GUI- und GUILayout nutzen	446
19.2	Performance	446
19.2.1	Rendering-Statistik	447
19.2.2	Batching-Verfahren	448
19.2.3	Analyse mit dem Profiler	450
19.2.4	Echtzeit-Analyse auf Endgeräten	451
19.2.4.1	Webplayer-Game remote analysieren	451
19.2.4.2	Android-Game remote analysieren	452
19.2.4.3	iOS-Game remote analysieren	452
20	Spiele erstellen und publizieren	453
20.1	Der Build-Prozess	453
20.1.1	Szenen des Spiels	454
20.1.2	Plattformen	455
20.1.3	Notwendige SDKs	455
20.1.4	Plattformspezifische Optionen	456
20.1.5	Developer Builds	456
20.1.5.1	Autoconnect Profiler	456
20.1.5.2	Script Debugging	456
20.2	Publizieren	457
20.2.1	App	458
20.2.2	Browser-Game	458
20.2.3	Desktop-Anwendung	459

21	Erstes Beispiel-Game: 2D-Touch-Game	461
21.1	Projekt und Szene	461
21.1.1	Die Kamera	463
21.1.2	Texturen importieren und Sprites definieren	463
21.2	Gespenster und Hintergrund	465
21.2.1	Gespenster animieren	468
21.2.2	Gespenster laufen lassen	472
21.2.3	Gespenster-Prefab erstellen	474
21.3	Der GameController	474
21.3.1	Der Spawner	475
21.3.2	Level-Anzeige	477
21.3.3	Der Input-Controller	478
21.3.4	Game Over-UI	480
21.3.5	Hintergrundmusik	486
21.4	Punkte zählen	487
21.5	Spielende	488
21.6	Spiel erstellen	489
22	Zweites Beispiel-Game: 3D Dungeon Crawler	491
22.1	Level-Design	492
22.1.1	Modellimport	493
22.1.1.1	Allgemeine Import Settings	493
22.1.1.2	bat_03 Import Settings	493
22.1.1.3	crate_01 Import Settings	494
22.1.1.4	stone_01 Import Settings	494
22.1.1.5	floor_01 Import Settings	494
22.1.2	Materials konfigurieren	494
22.1.3	Prefabs erstellen	495
22.1.3.1	Wall Prefab	495
22.1.3.2	Torch Prefab	495
22.1.3.3	Wall_Torch Prefab	496
22.1.3.4	Floor Prefab	496
22.1.3.5	Ceiling Prefab	497
22.1.4	Dungeon erstellen	497
22.1.4.1	Boden erstellen	497
22.1.4.2	Wände erstellen	498
22.1.4.3	Decke erstellen	499
22.1.4.4	Allgemeine Licht-Einstellungen	499
22.1.4.5	Fackelbeleuchtung	499
22.1.4.6	Akzent-Beleuchtung	501
22.1.5	Dekoration erstellen	502
22.1.5.1	Crate Prefab	502
22.1.5.2	Barrel Prefabs	502

22.2	Inventarsystem erstellen	503
22.2.1	Verwaltungslogik	503
22.2.1.1	InventoryItem.cs	507
22.2.1.2	CreateInventoryItem.cs	507
22.2.1.3	Inventory.cs	508
22.2.1.4	PickableItem.cs	510
22.2.2	Oberfläche des Inventarsystems	510
22.2.3	Inventar-Items	514
22.2.3.1	HoverEffects.cs	519
22.3	Game Controller	520
22.4	Spieler erstellen	521
22.4.1	Lebensverwaltung	522
22.4.1.1	LifePointController.cs	530
22.4.1.2	HealthController.cs	531
22.4.1.3	PlayerHealth.cs	531
22.4.2	Spielersteuerung	533
22.4.2.1	PlayerController.cs	538
22.4.2.2	Footsteps.cs	539
22.4.2.3	Shooting.cs	540
22.4.3	Wurfstein entwickeln	541
22.4.3.1	StoneBehaviour.cs	544
22.4.3.2	Zerberstenden Stein konfigurieren	545
22.4.4	Lautstärke steuern	547
22.5	Quest erstellen	548
22.5.1	Erfahrungspunkte verwalten	548
22.5.1.1	EPController.cs	549
22.5.2	Questgeber erstellen	550
22.5.2.1	WaterQuest.cs	555
22.5.2.2	WaterdropSound.cs	557
22.5.2.3	InGameMenu.cs	557
22.5.3	Sub-Quest erstellen	558
22.5.3.1	AnimateDoor.cs	562
22.6	Gegner erstellen	564
22.6.1	Model-, Rig- und Animationsimport	564
22.6.2	Komponenten und Prefab konfigurieren	565
22.6.3	Animator Controller erstellen	567
22.6.4	NavMesh erstellen	569
22.6.5	Umgebung und Feinde erkennen	570
22.6.5.1	EnemySonar.cs	571
22.6.6	Gesundheitszustand verwalten	572
22.6.6.1	EnemyHealth.cs	575
22.6.7	Künstliche Intelligenz entwickeln	576
22.6.7.1	EnemyAI.cs	582

22.7	Eröffnungsszene	585
22.7.1	Szene erstellen	585
22.7.2	Startmenü-Logik erstellen	586
22.7.2.1	MainMenu.cs	587
22.7.3	Menü-GUI erstellen	588
22.8	Web-Player-Anpassungen	590
22.8.1	Web-Player-Template ändern	590
22.8.2	Quit-Methode im Web-Player abfangen	590
22.9	Finale Einstellungen	591
22.10	So könnte es weitergehen	594
23	Der Produktionsprozess in der Spieleentwicklung	595
23.1	Die Produktionsphasen	595
23.1.1	Ideen- und Konzeptionsphase	596
23.1.2	Planungsphase	596
23.1.3	Entwicklungsphase	596
23.1.4	Testphase	597
23.1.5	Veröffentlichung und Postproduktion	597
23.2	Das Game-Design-Dokument	597
24	Schlusswort	599
Index	601