

BASISCURSUS

GAME DESIGN

**EXTRA:
MET 3 GAME-
DESIGNS**

MERIJN DE BOER



A C A D E M I C S E R V I C E

Basiscursussen verschenen bij Academic Service:

Basiscursus Access 2010
Basiscursus Access 2007
Basiscursus Access 2003
Basiscursus Access 2002
Basiscursus Apps ontwikkelen
Basiscursus ASP.NET
Basiscursus AutoCAD 2013 en LT 2013
Basiscursus AutoCAD 2012 en LT 2012
Basiscursus AutoCAD 2011 en LT 2011
Basiscursus AutoCAD 2010 en LT 2010
Basiscursus AutoCAD 2009 en LT 2009
Basiscursus AutoCAD 2008 en LT 2008
Basiscursus AutoCAD 2007 en LT 2007
Basiscursus AutoCAD 2005 en LT 2005
Basiscursus AutoCAD 2004
Basiscursus AutoCAD LT 2004
Basiscursus C++ 3e herziene druk
Basiscursus Cascading Style Sheets
Basiscursus Contribute
Basiscursus Dreamweaver CS5
Basiscursus Dreamweaver CS4
Basiscursus Dreamweaver CS3
Basiscursus Dreamweaver 8
Basiscursus Dreamweaver MX 2004
Basiscursus Dreamweaver MX
Basiscursus Drupal
Basiscursus Excel 2010
Basiscursus Excel 2007
Basiscursus Excel 2003
Basiscursus Excel 2002
Basiscursus Flash CS4
Basiscursus Flash CS3
Basiscursus Flash 8
Basiscursus Flash MX 2004
Basiscursus Flash MX
Basiscursus Flash ActionScript
Basiscursus FrontPage 2003
Basiscursus FrontPage 2002
Basiscursus Gamedesign
Basiscursus HTML 5
Basiscursus HTML 4.01
Basiscursus Illustrator CS4
Basiscursus Illustrator CS3
Basiscursus Illustrator CS2
Basiscursus Illustrator 10/CS
Basiscursus InDesign CS4
Basiscursus InDesign CS3
Basiscursus InDesign CS2
Basiscursus InDesign CS
Basiscursus Internet, 3e herziene druk
Basiscursus Internet Explorer 6
Basiscursus Java, 2e herziene druk
Basiscursus JavaScript 1.5
Basiscursus Joomla! 3
Basiscursus Joomla! 1.6
Basiscursus Joomla! 1.5
Basiscursus Mac OSX 10.5 Leopard
Basiscursus Mac OSX 10.3 Panther
Basiscursus Mac OSX 10.2 Jaguar

Basiscursus Office 365
Basiscursus Outlook 2010
Basiscursus Outlook 2007
Basiscursus Outlook 2003
Basiscursus Outlook 2002
Basiscursus Paint Shop Pro X
Basiscursus Paint Shop Pro 9
Basiscursus Photoshop Elements 3.0
Basiscursus Photoshop CS5
Basiscursus Photoshop CS4
Basiscursus Photoshop CS3
Basiscursus Photoshop CS2
Basiscursus Photoshop CS
Basiscursus Photoshop 7
Basiscursus PHP 6
Basiscursus PHP5.4 en MySQL
Basiscursus PHP 5
Basiscursus PHP 4.2
Basiscursus Pinnacle Liquid Edition
Basiscursus Pinnacle Studio 10
Basiscursus Pinnacle Studio 9
Basiscursus PowerPoint 2010
Basiscursus PowerPoint 2007
Basiscursus PowerPoint 2003
Basiscursus PowerPoint 2002
Basiscursus Premiere Elements
Basiscursus Premiere Pro
Basiscursus Premiere 6.5
Basiscursus Project 2003
Basiscursus QuarkXPress 5
Basiscursus SEO & HTML5
Basiscursus SQL, 2e herziene druk
Basiscursus SUSE Linux 10
Basiscursus Ubuntu
Basiscursus Ubuntu 10.04
Basiscursus Visio 2010
Basiscursus Visio 2002
Basiscursus Visual Basic 2005 EE
Basiscursus Visual Basic.NET
Basiscursus Visual Basic 6.0
Basiscursus Windows 7
Basiscursus Windows 8
Basiscursus Windows Vista
Basiscursus Windows XP
Basiscursus Word 2010
Basiscursus Word 2007
Basiscursus Word 2003
Basiscursus Word 2002
Basiscursus XHTML 1.0
Basiscursus XML herziene editie

Voor meer informatie en bestellingen:

Sdu Klantenservice
Postbus 20014
2500 EA Den Haag
Tel.: 070-3789880
Website: <http://www.academicservice.nl>

Merijn de Boer

Basiscursus Gamedesign



Meer informatie over deze en andere uitgaven kunt u verkrijgen bij:

Sdu Klantenservice
Postbus 20014
2500 EA Den Haag
tel.: (070) 378 98 80
www.sdu.nl/service

Omslagontwerp: LandGraphics, Amsterdam
Vormgeving: Redactie bureau Ron Heijer, Markelo
Zetwerk: Fritschy opmaak & redactie, Leiden
Druk- en bindwerk: Wilco, Amersfoort

ISBN: 978 90 12 58538 5
NUR: 980

© Sdu Uitgevers 2013

Alle rechten voorbehouden. Alle intellectuele eigendomsrechten, zoals auteurs- en databankrechten, ten aanzien van deze uitgave worden uitdrukkelijk voorbehouden. Deze rechten berusten bij Sdu Uitgevers bv en de auteur.

Behoudens de in of krachtens de Auteurswet gestelde uitzonderingen, mag niets uit deze uitgave worden verveelvoudigd, opgeslagen in een geautomatiseerd gegevensbestand of openbaar gemaakt in enige vorm of op enige wijze, hetzij elektronisch, mechanisch, door fotokopieën, opnamen of enige andere manier, zonder voorafgaande schriftelijke toestemming van de uitgever.

Voor zover het maken van reprografische verveelvoudigingen uit deze uitgave is toegestaan op grond van artikel 16 h Auteurswet, dient men de daarvoor wettelijk verschuldigde vergoedingen te voldoen aan de Stichting Reprorecht (Postbus 3051, 2130 KB Hoofddorp, www.reprorecht.nl). Voor het overnemen van gedeelte(n) uit deze uitgave in bloemlezingen, readers en andere compilatiewerken (artikel 16 Auteurswet) dient men zich te wenden tot de Stichting PRO (Stichting Publicatie- en Reproductierechten Organisatie, Postbus 3060, 2130 KB Hoofddorp, www.cedar.nl/pro). Voor het overnemen van een gedeelte van deze uitgave ten behoeve van commerciële doeleinden dient men zich te wenden tot de uitgever.

Hoewel aan de totstandkoming van deze uitgave de uiterste zorg is besteed, kan voor de afwezigheid van eventuele (druk)fouten en onvolledigheden niet worden ingestaan en aanvaarden de auteur(s), redacteur(en) en uitgever deswege geen aansprakelijkheid voor de gevolgen van eventueel voorkomende fouten en onvolledigheden.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the publisher's prior consent. While every effort has been made to ensure the reliability of the information presented in this publication, Sdu Uitgevers neither guarantees the accuracy of the data contained herein nor accepts responsibility for errors or omissions or their consequences.

Inhoud

Inleiding 7

Hoofdstuk 1 Hoe maak je een goede start? 9

- 1.1 Voor wie is dit boek bedoeld? 11
- 1.2 Het ontwikkelen van een idee 12
- 1.3 Het vinden van de juiste software 14
- 1.4 Construct 2: de basis voor dit boek 19
- 1.5 De eerste obstakels 21
- 1.6 De kunst van de eenvoud: klein beginnen 22
- 1.7 Van ambitie naar gameontwikkelaar: een persoonlijke noot 23
- 1.8 Interview: Launching Pad Games: succesvolle studio is even op 'break' 25

Hoofdstuk 2 Inspiratie en grafisch materiaal 31

- 2.1 Voorbeelden in Construct 2 32
- 2.2 Hulp op forums en websites 33
- 2.3 Gratis kunst voor je games 34
- 2.4 Gratis geluid voor je games 35

Hoofdstuk 3 Construct 2: snel kant-en-klare games maken 37

- 3.1 Wat is eigenlijk een engine? 38
- 3.2 Welk type games maak je met Construct 2? 39
- 3.3 Construct 2: een eerste indruk 40
- 3.4 De lay-out van het scherm 42
- 3.5 Een achtergrond maken voor Construct 2 47
- 3.6 Diepte in een spel brengen met lagen 50
- 3.7 Objecten tot leven brengen 51
- 3.8 Tips voor het werken met Construct 2 54

Hoofdstuk 4 Klassieke spellen uit de oude doos 57

- 4.1 Cheat sheets: jouw kant-en-klare gamedesigns 57
- 4.2 Pong: klassieke game in 15 minuten 58
- 4.3 Variatie op Pong: Breakout 72
- 4.4 Asteroids: zelfgetekende meteorieten knallen! 94

Hoofdstuk 5 Andere games maken 113

- 5.1 Platformspellen 113
- 5.2 Top down shooters 114
- 5.3 Interesse gewekt? Volg een gamestudie! 114
- 5.4 Interview: ervaringen van een student gamedesign 116

Hoofdstuk 6 Hoe maak je je game bijzonder? 123

- 6.1 Praktijkvoorbeelden uit digitale app-winkels 124
- 6.2 Interview: Vlambeer Games 129

Hoofdstuk 7 Professionele games maken en geld verdienen 135

- 7.1 Exporteren vanuit Construct 2 135
- 7.2 Publiceren op het web 136
- 7.3 App stores van Apple en Google 137

Hoofdstuk 8 Games afstemmen op het type platform 139

- 8.1 Touch: besturing op een aanraakscherm 140
- 8.2 Besturingsknoppen op het scherm plaatsen 142
- 8.3 Welke besturing past bij welk spel? 146

Nawoord 147

Register 149

Inleiding

Dit boek is een basiscursus gamedesign en daarbij zijn twee zaken erg belangrijk: *basis* en *design*. We gaan geen tien verschillende soorten games maken of stappenplannen doorlopen waarna je denkt 'en hoe begin ik nu zelf dan?'. Het hele idee achter dit boek is het ontwerp van een game en de basisbehoeften.

Wat betekent dit? Dat je geen honderden voorbeelden of opdrachten voorgeschoteld krijgt, maar met een gericht idee aan de slag kunt gaan. Uiteraard leer je hoe je bepaalde soorten spellen maakt, maar wat veel belangrijker is: hoe je met de minste moeite zo'n game interessant maakt en hoe je een keuze maakt voor het type spel dat je wilt produceren.

Belangrijk daarbij zijn de voorbeelden uit de praktijk, rechtstreeks van succesvolle gameontwikkelaars die voor dit boek zijn geïnterviewd. Je vindt de verslagen verspreid door het boek. Wil je snel aan de slag met het maken van spellen, dan blader je meteen naar hoofdstuk 4, waar je de formules voor 's werelds eerste game-klassiekers treft. Dit boek hoeft je zeker niet helemaal door te pluizen om meteen aan de slag te gaan, maar het is wel aan te raden. Lees je het goed door, dan begrijp je hoe bepaalde verhoudingen en afspraken werken, waardoor je alles kunt combineren in je eigen game.

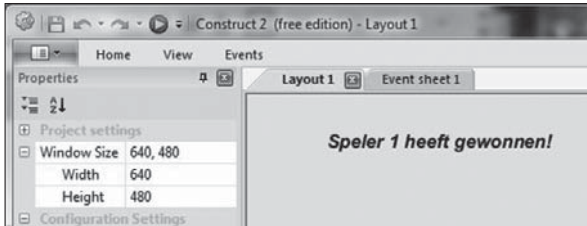
Bedenken wat je wilt is lastig, omdat je je meteen een ideeëndief voelt als je gaat kijken naar andere games. Dat is helemaal niet erg, want het wiel opnieuw uitvinden is juist bij het maken van spellen helemaal geen goed idee. Werk daarom meer vanuit de gedachte dat je iets moois wilt maken dat je zelf ook zou willen spelen of dat je vrienden of familie helemaal te gek zouden vinden.

In dit boek lees je over de verschillende technieken in Construct 2 die ervoor zorgen dat dingen bewegen, zodat je bereikt wat je wilt met je eerste ingeving. Uiteraard begint het allemaal bij een idee en daarom is het belangrijk dat je weet wat je wilt. Wie zijn de hoofdpersonen in het spel en wat is het doel? Ook is het belangrijk dat de speler wordt beloond met punten, nieuwe speelbeurten of gewoon een prettige speelervaring.

Tegenwoordig zijn er veel programma's beschikbaar die alle noodzaak voor programmeren weghalen. Veel van deze software, zoals Construct 2, is gratis en werkt volledig met het principe van slepen en neerzetten (*drag and drop*). Dit betekent dat jij je maar met drie zaken hoeft bezig te houden: het plaatsen van objecten in de lay-out van Construct 2, het toewijzen van gedrag aan

die objecten en het arrangeren van botsingen en gebeurtenissen, die ‘*events*’ heten in de software-taal.

Dat laatste is waar het uiteindelijk om draait. Als ware de muiscursor een toverstokje, sprankel je in het tabblad *Events* de magie over de objecten, waardoor ze tot leven komen. In de browser check je snel of de onderdelen (helden, vijanden, etc.) hun afspraken (de geplande *events*) nakomen.



Figuur 1 In het tabblad *Layout* van Construct 2 plaats je de achtergrond en spelonderdelen, in het tabblad *Events* programmeer je gebeurtenissen

In dit boek zal het je opvallen dat programmeren een heel andere betekenis krijgt. Vergeet codes, formules en schermen met alleen maar cijfers en teksten: Construct 2 maakt van al deze code een visuele weergave die voor iedereen te begrijpen is. Programmeren is vanaf dit moment het voorbereiden van een kamer (de lay-out), het koppelen van gedrag aan objecten (met de functie *behavior*) en het laten bewegen van alle acteurs en onderdelen in het spel.

Het gebruik van dit boek

Namen van knoppen, schermonderdelen, opties en dialoogvensters worden *in dit lettertype* weergegeven. Toetsen die ingedrukt moeten worden staan aangegeven met de gebruikelijke naam, zoals Shift of Ctrl. Ze worden in de tekst **vet** gedrukt weergegeven.

Regelmatig zijn in dit boek waarschuwingen, tips en opmerkingen te lezen. Deze zijn te herkennen aan de volgende icoontjes:



= waarschuwing,



= tip,



= opmerking.

1 Hoe maak je een goede start?

Het hebben van een leuk idee voor een game is uiteraard belangrijk, maar dan? Hoe ga je het idee uitproberen en uiteindelijk ten uitvoer brengen? Hoe maak je de plaatjes of digitaliseer je je eigen schetsen? Een ding is zeker: programmeren in de klassieke betekenis van het woord is niet nodig. Dankzij programma's als Construct 2 krijg je de mogelijkheid om alles wat je bedenkt gewoon in het scherm te slepen en met heldere knoppen functies te koppelen aan de onderdelen van je lay-out (ofwel het *level*).

Je werkt in Construct 2 alleen maar met slepen en plakken en hoeft bijna niets in te voeren. In feite programmeer je dus op een volledig andere wijze, zonder het invoeren van regels met code. Programmeren is hier arrangeren, dirigeren, organiseren en daar hoeft geen letter voor te worden getypt. Zelfs gamedesigners gebruiken geregeld dit soort software voor hun games, simpelweg omdat dit ze honderden regels code bespaart. Een stuk software voor het slepen en plakken van objecten, achtergronden en het kiezen van manieren waarop de 'spelers' botsen, is de sleutel bij het maken van een eerste spel. Vooral als je helemaal geen programmeerervaring hebt.

In de volgende hoofdstukken lees je hoe je een idee ontwikkelt en een concept stap voor stap test in Construct 2. Ook lees je over de mogelijke problemen die opdoemen zodra je een interessant idee hebt. Wat redelijk snel duidelijk wordt is de kracht van eenvoud; een speltype werkt het beste als het voor zowel jou als de speler simpel én vermakelijk is. Er zijn niet zo gek veel regels verbonden aan het interessant maken van een spel, maar één afspraak geldt altijd: er moet iets te winnen zijn.

Winst en het idee van '*instant gratification*' (direct plezier/vermaak) zijn vooral belangrijk bij games op Facebook of mobiele apparaten. Het zijn de gemakkelijkste spellen om te maken, omdat verhaal, graphics en geluid er vrijwel niet toe doen. Wat veel belangrijker blijkt, is het neerzetten van een verslavend spelletje dat mensen overhaalt om door te blijven spelen (en wie weet kopen de spelers af en toe nieuwe onderdelen om sneller voortgang te boeken).

Een belangrijk nieuw speltype van de laatste paar jaren is het *freemium*-model. Dit type game lijkt gratis en is vaak ook voor een groot gedeelte kosteloos te spelen. Wil je echter verder geraken of meer punten scoren (om mee te pronken op Facebook bijvoorbeeld), dan is een aankoop vereist.

Het type spel is nogal in opspraak geraakt de laatste paar jaren, bijvoorbeeld in het geval van de Smurfen. Kort nadat dit spel (*Smurfs' Village*) op de iPhone verscheen, zagen ouders honderden, soms duizenden euro's verdwijnen van een bankrekening, via hun creditcard. Het bovenstaande spel, dat draait om het bouwen van huisjes en kweken van gewassen, werkt als volgt: je plant iets gratis maar moet dan soms uren wachten totdat het gewas kan worden geoogst. Zo kun je per dag misschien een aantal pompoenen kweken en wat aardappels rooien. Maar wil je meteen je fruit in de mand werpen dan moet je een joker inzetten: de smurfberry.

Wat bleek na verloop van tijd? Na het kopen van smurfberrys door ouders voor hun kroost, begonnen kinderen al snel zelf de 'upgrade' te kopen, omdat het creditcardnummer niet opnieuw hoefde te worden ingevoerd. Een grote fout van Apple, want het kostte een aantal gezinnen duizenden euro's.



Figuur 1.1 Freemium games, waarin je sneller voortgang boekt na betalingen, raakten snel in opspraak. In het geval van *Smurfs' village* kostte het talloze ouders veel te veel geld. Voor de ontwikkelaar was het smurfenspel echter een goudmijn: *Smurfs' Village* is het jarenlang het meest opbrengende spel in Apple's App Store geweest

Voordat je echter aan de slag gaat met games die geld in het laadje brengen, is het handig om eerst naar de basis te kijken. Wat zijn je eigen vaardigheden en hoe pas je deze het beste toe bij het maken van een eerste spel?

Dit boek legt aan de hand van klassieke games uit hoe Construct 2 werkt en hoe je helemaal zonder programmeren in de klassieke zin (code en programmeren) een spel maakt. De spellen die je maakt vanaf hoofdstuk 4 lijken wellicht simpel, maar ze verklaren op speelse wijze hoe mechanieken, ook wel *mechanics* genoemd, werken.

De drie spellen in dit boek zijn sterk met elkaar verweven, omdat ze de basisprincipes van Construct 2 uit de doeken doen en elkaar aanvullen met het oog op alternatieve werkmethodes. Zo merk je dat je op verschillende manieren objecten kunt laten bewegen en een game op diverse wijzen kunt beëindigen. Heb je de drie spellen afgerond met de lessen in hoofdstuk 4, dan heb je een goed idee van de software en kun je zelf aan de slag met je eigen idee.

1.1 Voor wie is dit boek bedoeld?

Deze Basiscursus Gamedesign is bedoeld voor iedereen die ooit wilde beginnen met het maken van een game, maar niet wist waar of hoe. Dit boek levert de basisbeginselen met dank aan de software van Construct 2. Het programma biedt vrijwel de eenvoudigste manier om met slepen en plakken een volledig spel te maken, zonder wat voor programmeerervaring dan ook. Wil je daarna overstappen op een ander pakket, zoals het vergelijkbare Game Maker, dan is deze basis ook erg zinvol.

Je hebt nu een *basiscursus* in handen, wat betekent dat de materie is teruggebracht tot de eenvoud. Bovendien heeft de cursus waardevolle extraatjes voor je in petto.

De interviews geven je bijvoorbeeld een goed beeld van de industrie en hoe je daarin terechtkomt (mocht je dat willen). Je leest de verhalen van zowel doorgewinterde ontwikkelaars als kleine beginnende teams én zelfs de ervaringen van een student gamedesign. In een notendop is dit boek een uitkomst voor:

- iedereen die een eerste game wil maken voor het net, een mobiel apparaat of voor Windows;
- leerlingen die een multimediaal project op touw willen zetten;
- beginnende game-designers die sneller willen werken en wel wat trucjes kunnen gebruiken.

Het boek is dus niet alleen voor compleet onervaren designers bedoeld. Ook professionele gameontwikkelaars hebben baat bij Construct 2, omdat het de manier van werken versnelt. Programmeren in de klassieke zin is immers niet meer nodig, waardoor de designer meer tijd heeft voor het maken van plaatjes en het uitwerken van het game-idee.

De Basiscursus Gamedesign biedt een goede introductie voor Construct 2 en neemt een hoop gespeur op forums weg. Het boek laat duidelijk zien hoe je gedrag koppelt aan objecten (helden, vijanden) en hoe je de spelregels van je project gemakkelijk realiseert. Heb je al een goed idee voor een spel? Dan kun je na dit boek meteen aan de slag, zonder gedoe met honderden lessen en zonder het afvuren van talloze vragen op forums.

Uiteraard staan er ook een paar lessen voor je klaar in dit boek. Deze zijn echter bedoeld om je de trucjes van Construct 2 te leren en niet om je bezig te houden of je te laten leren door kopiëren. Je gaat geen ellenlange projecten starten en krijgt geen stampwerk voor de kiezen: het zijn de details per game die tellen. Hoe laat je een bal stuiteren tussen twee batjes? Hoe lok je vijanden naar je held? Waarom werkt die explosie niet, terwijl je toch echt het plaatje naar het scherm had gesleept? Het zijn de kleine spelregels van een *engine* (de motor die het spel aandrijft) die je onder de knie moet krijgen. Dat is precies wat er gebeurt tijdens het lezen van dit boek.

Alle projecten in deze basiscursus bevatten weer andere basiselementen van een bepaald soort game. Hoe verder je vordert in het boek (met name vanaf hoofdstuk 4), des te meer kun je uiteindelijk toevoegen aan elk spel. Vrijwel alle nieuwe regels die je leert, zijn weer toe te passen op vorige games. Zo voegt elk nieuw project waarde toe aan wat je al had geleerd.

Elk project blijft echter lekker gemakkelijk en toegankelijk. Elk spel is onafhankelijk te maken, zodat je ook meteen met een groter project aan de slag kunt gaan. Vanwege de spectaculaire effecten, menu's en uitgebreide gameplay kost een verder gevorderd project natuurlijk wel meer tijd en aandacht.

1.2 Het ontwikkelen van een idee

Net als bij een klassiek bordspel of gezelschapsspel moet je met wel heel erg revolutionaire ideeën op de proppen komen wil je hét nieuwe succesverhaal van het jaar maken. Bij het nadenken over je type spel moet je dus vooral naar de gevestigde orde kijken en overwegen wat voor jou haalbaar is om te maken.

Een paar tips vooraf:

- Begin met het ontwikkelen van een idee en een speltype, niet met het tekenen van je favoriete dieren/helden omdat je die zo cool vindt.
- Maak je geen zorgen over missende vaardigheden; verderop in het proces kun je altijd nog putten uit talenten om je heen.
- Zoek niet ellenlang naar gratis plaatjes als je een idee hebt, maar begin direct met Construct 2 en een paar blokjes.
- Speel met een aantal speltypes en kies er uiteindelijk één die je helemaal wilt uitwerken.

In dit boek lees je over verschillende manieren voor het maken van een game. Mocht je namelijk moeite hebben met het opzetten van een spel, dan kun je altijd ideeën lenen van anderen.



Denk maar aan Rovio, de makers van *Angry Birds*. Hun recentere spel *Amazing Alex* is volledig aangekocht en heette daarvoor *Casey's Contraptions*. Die game was niet veel eerder uitgekomen en was gemaakt door een tweekoppig team. Later verdween het ineens uit de App Store. Waarom? Omdat Rovio het recht op het spel had gekocht, een likje verf eroverheen gooide en het opnieuw uitbracht.

Misschien vraag je je af waarom Rovio na het grote succes van *Angry Birds* een game van twee onafhankelijke ontwikkelaars kocht. Het antwoord is lastig te geven, maar wellicht wilde het bedrijf af van het stempel 'ontwikkelaar van vogelspelletjes'. Toch is het jammer dat maar weinig mensen weten dat *Amazing Alex* vroeger *Casey's Contraptions* heette en helemaal door twee mensen is gemaakt.

Zo zie je wat er kan gebeuren als je een succesformule in handen hebt: is het goed genoeg, dan koopt een grotere ontwikkelaar het misschien van je. Het grappigste aan dit hele verhaal is het feit dat ook *Casey's Contraptions* geleend is van een ander spel: *The Incredible Machine* uit 1993!

Heb je dus een goed idee, dan kan het maar zo gebeuren dat een groot bedrijf het koopt, waardoor jij je kunt richten op nieuwe spellen of zelfs mensen kunt gaan aannemen in je nieuwe studio. Het kan ook anders lopen, wat kleine ontwikkelaar Nimblebit aan den lijve ondervond.

Nimblebit maakte een vermakelijk spel, *Tiny Tower*, waarin je appartementen bouwt in een grote toren. Het spel was ook kosteloos erg goed te spelen en vereiste alleen van de échte verslaafden periodieke betalingen. Apple koost *Tiny Tower* als spel van het jaar in 2011.

Sociale-gamesmagnaat Zynga kopieerde tot op het puntje van de neus van de hoofdrolspelers het gehele spel en noemde het *Dream Heights*. Nimblebit schreef Zynga een soort felicitatietweet met vergelijkingen tussen de spellen. *Dream Heights* was inderdaad volledig gekopieerd, maar wat kon de kleine ontwikkelaar doen? Niet veel, bleek later, want een type spel is erg moeilijk te patenteren. Heb je een revolutionair idee, dan kan het dus gebeuren dat een ander ermee aan de haal gaat.



Figuur 1.2 *Tiny Tower* van Nimblebit (toen met 3 medewerkers) werd volledig gekopieerd door Zynga (toen met 2.789 medewerkers). De ontwikkelaar kon niets doen tegen de gamesgigant, die toch echt plagiaat had gepleegd

Plagiaat is een veelvoorkomend probleem in de gameswereld. Zo kopieerde zelfs iemand het hele webspel *Triple Town* om het te verkopen voor mobiele apparaten. De ontwikkelaar was zelf ook met een mobiele versie bezig en kreeg het gelukkig voor elkaar om samen met Google haar eigen spel te verkopen in de Play Store (de spelletjeswinkel voor Android-telefoons).

In het geval van *Triple town* was het wel erg duidelijk dat er sprake was van plagiaat, omdat gewoon alle gameonderdelen gekopieerd waren en gebruikt in een mobiele versie. In hoofdstuk 2 lees je meer over het verzamelen van grafisch materiaal, maar dan legaal. Er is namelijk veel op het web te vinden

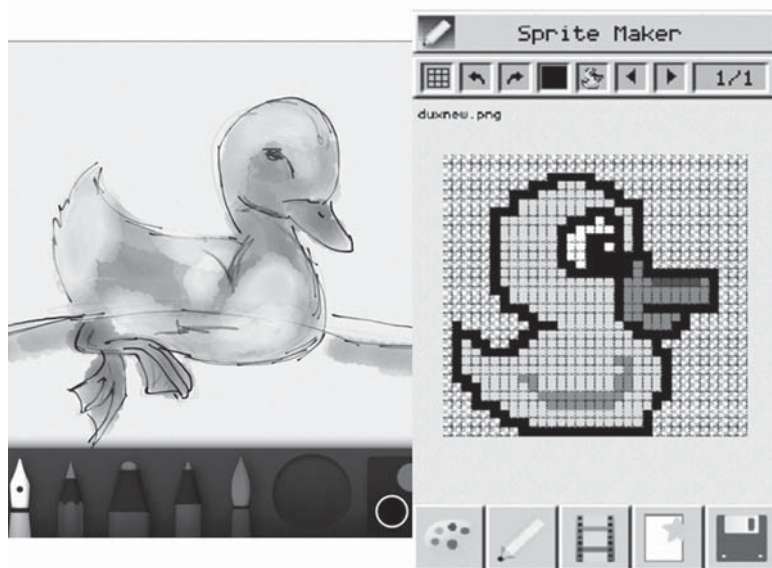
dat je gewoon mag gebruiken voor je eigen spel, wat de productie aanzienlijk versnelt.

1.3 Het vinden van de juiste software

Tegenover alle productieve softwarepakketten staat een gratis variant. Werk je graag met Adobe Photoshop, dan is GIMP een prima gratis alternatief. Werk je veel met Office, dan is LibreOffice een prachtige gratis versie. Ook voor het maken van games zijn veel gratis pakketten beschikbaar en in dit boek speelt Construct 2 een grote rol. Daarnaast is er natuurlijk een regenboog aan grafische programma's of zelfs websites beschikbaar waarmee je snel kunst voor je games in elkaar zet.

De meeste simpele games bestaan uit een achtergrond, spelonderdelen en in een later stadium de interface. Al deze onderdelen kun je zelf maken (of lenen en bewerken) en hiervoor zijn veel programma's beschikbaar. De gemakkelijkste manier om snelle tekeningen te maken is met een pixel-editor. Zo'n programma stelt je in staat om klassiek ogende games te produceren, wat voor iedereen mogelijk is. Het is namelijk een soort trucje: je reduceert helden, vijanden en meer tot blokjes, wat er vrijwel altijd prima uitziet.

Ook al ben je een beeldend kunstenaar of begenadigd striptekenaar; je werk moet nog steeds van papier naar digitale vorm worden omgezet. Gelukkig zijn er veel manieren om dat te doen en bestaat zelfs de mogelijkheid om direct een digitale tekening te maken. Tablets zijn daar perfect voor. Krijg je de smaak écht te pakken en heb je aan het einde van dit boek tal van ideeën waarmee je aan de slag wilt gaan, dan is zo'n tablet niet eens een gekke investering.



Figuur 1.3 Een getekende afbeelding kost meer tijd dan een pixelvoorstelling. De tekening links kostte zo'n tien minuten, tegenover één minuutje voor de tekening rechts

Het grote probleem bij het tekenen van graphics is dus niet eens talent of tijd. Veel lastiger is het overzetten van een tekening naar digitale vorm. Want als je allemaal schetsen gemaakt hebt op papier, hoe zet je die dan om naar bruikbare objecten?

Het antwoord in het vroege stadium van je gameontwerp is simpel: gebruik geen bonte tekeningen maar gewoon pixels of vormen. In hoofdstuk 4 vind je het eerste spel en de volledige handleiding om het te maken. Onderdelen zijn bij dit spel (*Pong*) niet nodig, omdat de vierkantjes en blokjes voor dat klassieke spel allemaal vanuit Construct 2 worden getekend.

Tijdens het maken van het eerste spel wordt snel duidelijk wat de kracht is van Construct 2 en hoe belangrijk het tekenen van simpele blokjes is. Ga je voor de eerste keer een idee proefdraaien, dan is het niet belangrijk dat je held glimmende laarzen en wapperende haren heeft. Eerst moet de beweging kloppen, moeten alle toetsen werken en moet je idee al goed zijn uitgewerkt.

Pixelkunst op je pc of Mac

Dit boek gaat niet over grafisch design, dus een lijst van gratis grafische bewerkers is minder relevant. Misschien heb je een favoriete grafische bewerkers en ken je dat programma goed, dan is het fijn om hiermee te werken. Voor de basis van gamedesign gaan we echter in dit boek terug naar de basis van grafische elementen in games: pixels.



Figuur 1.4 *Superbrothers: Sword & Sworcery* is een game die inmiddels op smartphones, tablets, pc én Mac verkrijgbaar is. Het spel begon op mobiele apparaten maar werd zo populair dat het op elk type computer uitkwam

Pixels zijn min of meer de zichtbare punten in een digitale afbeelding. Vroeger zag je elk puntje, omdat de dichtheid van de pixels (ofwel de resolutie) erg laag

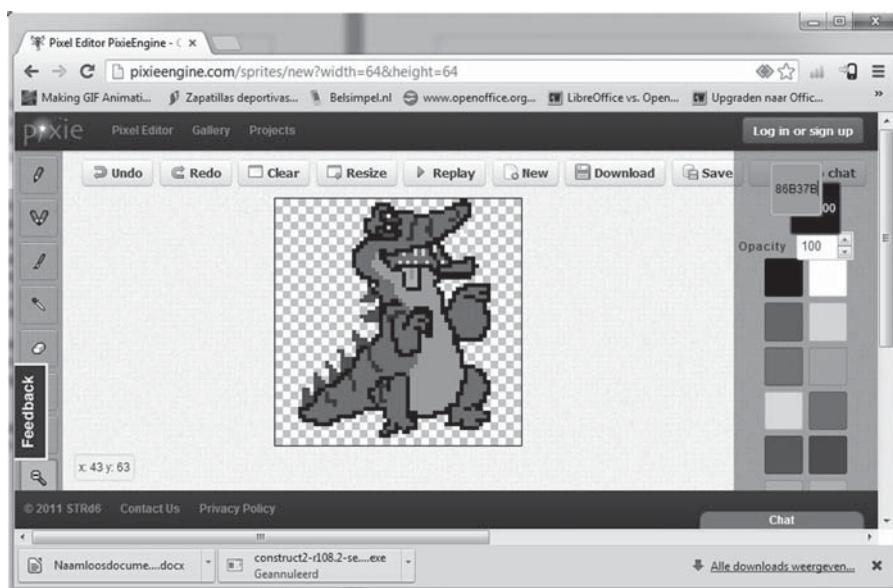
was. *Super Mario* leek een verzameling van blokjes, wat eigenlijk ook helemaal klopt. Dat blokkerige is weer helemaal hip en op mobiele apparaten krioelt het dan ook van de games met Pixel-art.

Pixelie-engine: online instrument voor prachtige plaatjes

Blokkerige helden en kruimelige achtergronden zijn via een simpele editor snel te maken en Pixelie-editor, een online dienst, is hier erg geschikt voor. Om de dienst te gebruiken ga je naar www.pixelie-engine.com/pixel-editor. De interface van de Pixel-editor lijkt op die van een grafische bewerkers als Photoshop of GIMP, met een belangrijk verschil: er is een symmetrisch potlood aanwezig.

Dankzij een symmetrisch potlood, dat al je lijnen spiegelt, is het mogelijk om supersnel objecten te maken. Dit werkt erg goed wanneer je een held van voren wilt tekenen of objecten met gelijke hoeken (zoals kisten, vloerpatronen, etc.).

Ook is het fijn dat de bewerkers standaard staat ingesteld op het maken van objecten van 32×32 pixels. Dit is een prima maat voor gameonderdelen, hoewel je uiteraard grotere onderdelen kunt maken. Alle onderdelen voor *Break-out* en *Asteroids* in hoofdstuk 4 zijn gemaakt met deze online bewerkers.



Figuur 1.5 Pixelie-engine is een prachtig online instrument voor het maken van gameonderdelen. De software gaat zelfs verder, want je kunt hele games maken vanuit je browser!



Voor de algemene website ga je naar <http://pixieengine.com>, wil je direct plaatjes maken, dan ga je naar <http://pixieengine.com/pixel-editor>.

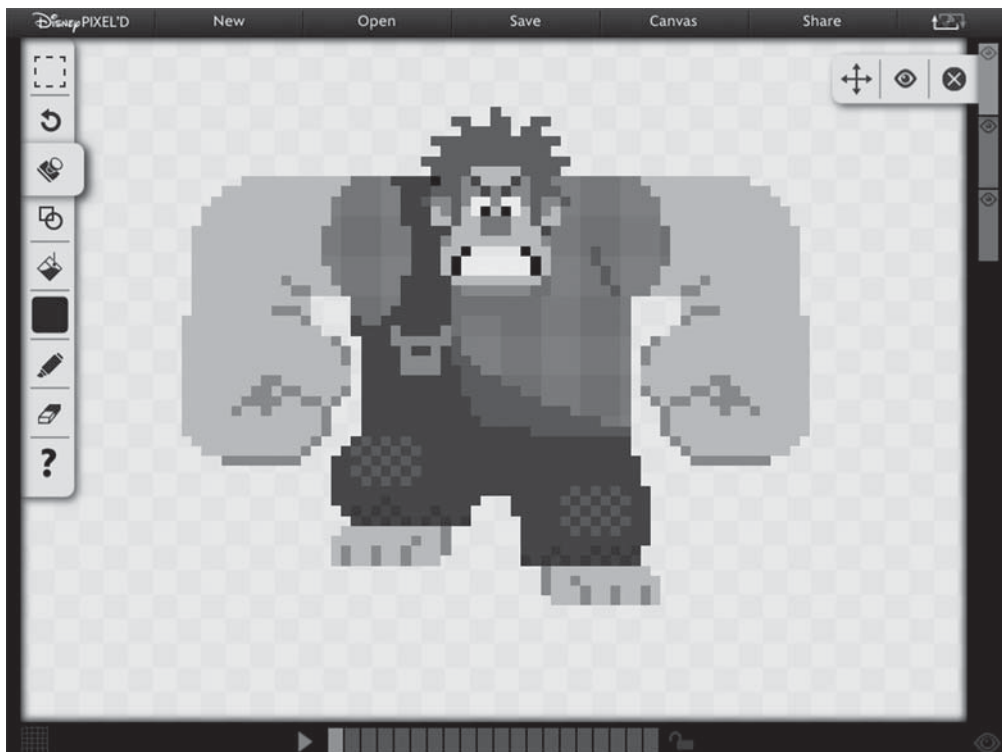
Pixelkunst op tablets

Heb je een iPad of andere tablet en wil je uiteindelijk games maken voor je tablet, dan zijn er veel goede programma's beschikbaar. Ik licht ter illustratie één programma voor iOS- (iPad-) en één programma voor Android-tablets toe.

Voor je iPad: Disney's creatieve bewerker Pixel'd

Voor Apple's iPad is er een prachtig stukje software beschikbaar uit de studio van Disney. *Pixel'd* is helemaal gratis en zit tjokvol voorbeelden uit Disney-films. Je treft er klassiekers als Mickey Mouse en Sneeuwwitje maar er zitten ook nieuwe karakters, bijvoorbeeld uit de film *Wreck it Ralf*, in. Waar het echter om gaat is de tekenfunctie: die is perfect en maakt zelfs animaties.

Pixel'd is gratis te downloaden in de App Store van Apple. Het programma is gemakkelijk in het gebruik, maar vereist wat oefening. Gelukkig zit er een handleiding bij de software, die stap voor stap uitlegt hoe alles werkt. Met *Pixel'd* maak je snel een plaatje of animatie, maar let op: de software zet een watermerk onder de afbeelding. In Construct 2 snijd je echter gemakkelijk een plaatje bij, waardoor dit geen enkel probleem vormt.



Figuur 1.6 Disney maakte voor de iPad een prachtig programma waarmee je in een handomdraai sprites kunt maken. Ook het maken van animaties is mogelijk



Om *Pixel'd* te downloaden zoek je in de App store naar de app. Het volstaat om in de winkel 'Pixel' in te vullen, want de app duikt dan snel genoeg op.

Heb je met *Pixel'd* een animatie of tekening gemaakt, dan stuur je deze gemakkelijk per mail naar jezelf om te gebruiken in Construct 2. In hoofdstuk 3 lees je meer over het importeren van plaatjes in Construct 2.

Voor Android-tablets: snel kunst maken met Sprite Maker

Heb je een Android-tablet, dan zijn er tal van programma's beschikbaar voor het maken van plaatjes. Een kleine waarschuwing vooraf: de apps zijn niet allemaal even goed. Sprite Maker is echter een aanrader; het is een simpel instrument waarmee je razendsnel onderdelen voor je spel tekent. De software is gratis te downloaden uit de Play Store en werkt bijzonder prettig.

Sprite Maker is bijzonder gebruiksvriendelijk, met een paar knoppen onder in het scherm en heldere tekenopties. In eerste instantie opent het programma een plaatje met 16x16 pixels (perfect voor gameonderdelen) maar dit is naar wens aan te passen.



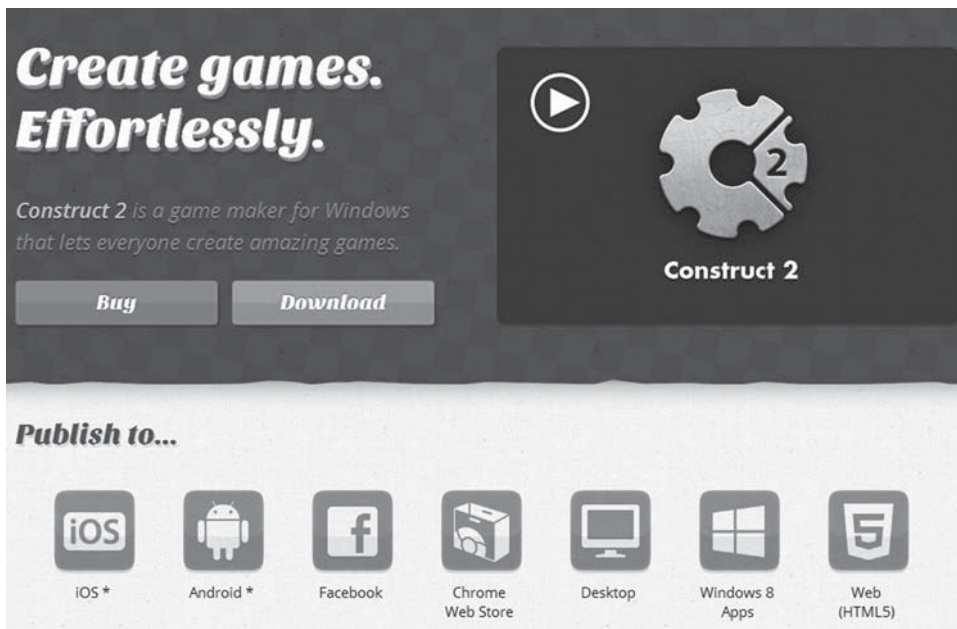
Figuur 1.7 *Sprite Maker* is een gebruiksvriendelijk programma voor Android-tablets en -telefoons. Met de software maak je snel onderdelen voor een spel, waaronder blokjes (voor de vloeren van platformspellen etc.) of blokkerige retro-helden voor je game

Hoewel bovenstaande programma's erg goed werken bij het maken van pixel-art, is het ook mogelijk om te putten uit een immense bron van gratis plaatjes en gameonderdelen. In hoofdstuk 2 lees je meer over het vinden van gratis materiaal voor je spellen; iets dat veel gevestigde ontwikkelaars geregeld doen. Het bespaart tijd en is zelfs een ode aan de kunst van anderen.

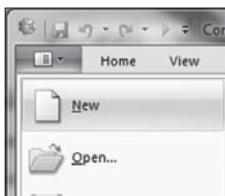
1.4 Construct 2: de basis voor dit boek

We eindigen dit hoofdstuk met de installatie van Construct 2: een van de gemakkelijkste programma's voor het maken van games. Je vindt de software via de website www.scirra.com en de basisversie van Construct 2 is helemaal gratis.

Construct 2 is gemaakt met Drag and Drop (slepen en neerzetten) als belangrijkste uitgangspunt. Met de software kun je zonder programmeerkennis en gebruik van code of formules aan de slag, want dat doet de software namelijk voor je. Jij hoeft alleen maar sprites (plaatjes/onderdelen) te maken, het idee te ontwikkelen en te ontdekken hoe je vanuit een speciaal tabblad (de *Event Sheet*) alles in beweging krijgt. Dit boek helpt je daarbij.



Figuur 1.8 Construct 2 download je van www.scirra.com, waarna je het in een mum van tijd installeert. Op de website zie je meteen voor welke platformen/besturingssystemen je spellen kunt maken



Figuur 1.9 Na installatie start je de software en klik je op de knop *File* (die met het pijltje linksbovenin). Selecteer dan *New* en Construct 2 slaat je hele spel in één bestand op, waarna een leeg scherm met twee tabbladen opent

Construct 2 werkt met de volgende tabbladen: *Layout* en *Event Sheet*. Het eerste blad is bedoeld voor de plaatsing van gameonderdelen, het tweede blad bevat alle gebeurtenissen op het scherm. Aan deze twee schermen ga je helemaal gewend raken na het lezen van dit boek, maar er is nog een derde ‘speler’ die de mogelijkheden van Construct 2 bepaalt: de functie *Behavior*.

Alle objecten die je in het programma gebruikt, zijn uit te rusten met een bepaald gedrag, dat varieert van rennen en springen (dit heet in Construct 2 toepasselijk *Platform*) tot het keihard wegvliegen in een bepaalde richting. Dat laatste gedrag betreft die van *Bullet*, ofwel een kogel.

Open je Construct 2 voor de eerste keer, dan oogt het vast een beetje intimiderend. De angst is van korte duur, want dubbelklik je op het hoofdscherm dan krijg je meteen de mogelijkheid om objecten te maken of te importeren. Heb je nog nooit een plaatje bewerkt of een sprite (gameonderdeel) gemaakt, dan is er geen man overboord: het eerste spel vereist geen enkele grafische voorkennis. Sterker nog: *Pong* (een van de eerste games voor spelcomputers) maak je volledig vanuit Construct 2.

Samenvattend bestaat Construct 2 dus uit drie onderdelen: de lay-out, de *Event Sheet* en de gedragsfunctie bij objecten. Een klein overzichtje:

Layout: de kamer met speeltjes

Hier begint het allemaal: het tabblad *Layout*. Een dubbele muisklik is het startsein voor je eerste stappen in de software, gevolgd door een scherm met alle mogelijke elementen van een kamer. Dat kunnen gewone objecten zijn die later gaan bewegen, springen en noem het maar op, tekstvakken die de score bijhouden of zelfs een besturingselement zoals een toetsenbord.

Event sheet: het commandocentrum voor gebeurtenissen

Vanuit dit tabblad voeg je een *event*, ofwel gebeurtenis toe aan het spel. Het zijn eigenlijk de spelregels die hier worden bepaald en deze hangen allemaal af van de kunstjes die je jouw objecten hebt meegegeven. Is één van je objecten uitgerust met het gedrag van een kogel, dan staan alle mogelijkheden van de kogel klaar in je *Event Sheet*. Heb je het besturingselement *Keyboard* toegevoegd in *Layout*, dan zijn al je objecten te koppelen aan de knoppen van je toetsenbord.

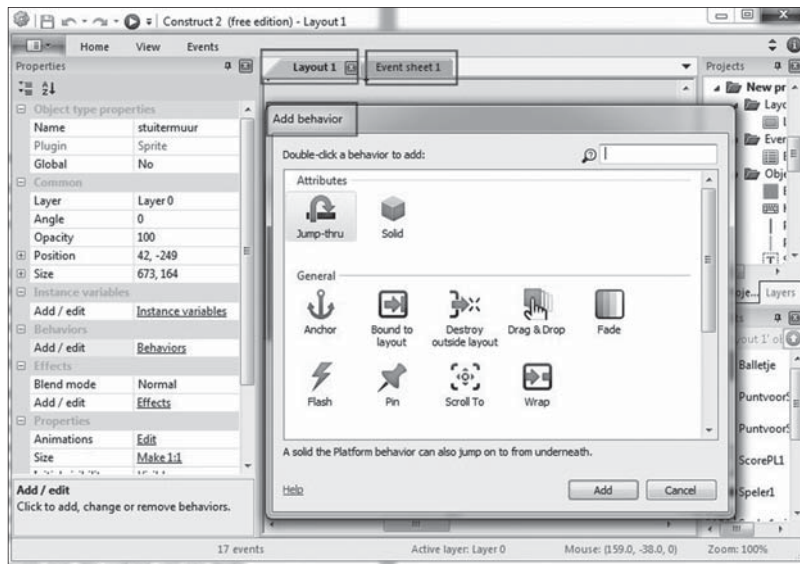
Behaviors: superkrachten voor je spelobjecten

Heb je een kamer (ofwel de lay-out) ingericht met spelonderdelen, dan wijs je deze (super)krachten of een bepaald gedrag toe met de functie *Behaviors*. Bij het aanklikken van elk object in het scherm verschijnen aan de linkerkant van het scherm de eigenschappen, waaronder ook de gedragingen, die erg belangrijk zijn. Of het nu gaat om het afvuren van kogels, het toevoegen van

zwaartekracht of het laten rondtollen van een meteoriet; het is allemaal mogelijk vanuit deze functie.



Met Construct 2 kun je maken wat je wilt, maar je moet wel weten hoe je een kamer vult met objecten en deze met elkaar laat samenwerken. Dit boek helpt je daarmee en legt haarfijn uit hoe je zonder moeilijke codes en formules met Construct 2 je eerste games maakt.



Figuur 1.10 De drie belangrijkste onderdelen van Construct 2: de *Layout*, de *Event Sheet* en *Behaviors*

1.5 De eerste obstakels

Zoals je misschien al hebt gemerkt, is de software die centraal staat in dit boek (Construct 2) nét niet voldoende voor het maken van een volledige (en super-coole) game. Je merkt al snel dat je echt wel plaatjes nodig hebt en een beetje moet gaan tekenen om aan de slag te kunnen. Uiteraard kun je gewoon beginnen met de voorbeeldgames in dit boek, maar wil je zelf iets maken dan moet je toch echt even rustig hiervoor gaan zitten en je ideeën uitwerken.

Pas dan ontstaan originele concepten, die als een lichtpeertje opduiken boven je hoofd en de passie voor gamedesign aanzwengelen. Het kopiëren van een game is niet zo moeilijk. Het kopiëren of maken van een succesvolle game is een heel ander verhaal.

Verderop in het boek lees je hoe je succesvolle formules voor games kopieert met behulp van events, ofwel virtuele weergaven van de broncode. Het mag duidelijk zijn dat dit niet gemakkelijk is, want een kopie vereist nieuwe

beelden en hoe gemakkelijk je ook werkt met Construct 2: alle gedragingen (van katapulten tot neerstortende vogels) dien je nog steeds te arrangeren.

Een succesvolle game is een product van lange voorbereiding. Denk hierbij aan:

- klaarstomen van grafisch materiaal;
- animeren van je helden, vijanden;
- maken van achtergronden;
- instellen van afspraken tussen gameonderdelen;
- testen van je spel;
- achter je hoofd krabben wanneer het fout gaat.

Ondanks de bovenstaande (ellenlange) processen krijg je echter vast een lekker gevoel wanneer vogels eindelijk uit je zelfgebouwde katapult vliegen, op weg naar hun doel.

1.6 De kunst van de eenvoud: klein beginnen

Heb je wilde ideeën voor het maken van games, dan mondt dit soms uit in hoge verwachtingen. Je denkt wellicht aan spectaculaire explosies, vloeiend geanimeerde helden of prachtige handgeschilderde achtergronden (als je creatief bent aangelegd).

Tegenwoordig zijn het echter de mechanismen in een game die het succes bepalen. De grafische kant is vrijwel altijd ondersteunend voor de manier waarop een game de speler binnentrekt (en niet meer loslaat).

Dit boek is gericht op de eerste grondbeginselen, waarna je écht aan de slag kunt gaan. We beginnen klein in hoofdstuk 4, maar ben je klaar met het eerste spel, dan heb je iets bereikt waarvoor in 1972 een heel team vereist was. Hoe eenvoudig het eerste spel ook lijkt, je bent in staat om in je eentje dingen te laten bewegen en een volledige spelervaring neer te zetten. Dat is een vrij unieke vaardigheid, waarvandaan je nóg mooiere spellen ontwikkelt.

Bovendien leer je bij elk spel uit dit boek nieuwe onderdelen toe te passen, zodat de speelervaring steeds completer wordt. Uiteindelijk combineer je het geleerde en weet je (beter) hoe je dat ambitieuze idee van je ten uitvoer kunt brengen.

Een game met een goed concept kun je uiteindelijk misschien verkopen, wat bij een spel als *Angry Birds* min of meer is gebeurd. Rovio, het ontwikkelteam, bracht het eerste spel uit de serie uit bij een kleine uitgever in het Verenigd Koninkrijk, maar publiceerde daarna al haar spellen zelf. Chillingo, een kleine uitgever, werd kort na publicatie een van de grootste uitgevers van mobiele spellen en later zelfs door EA (Electronic Arts) gekocht voor bijna 20 miljoen dollar.



Figuur 1.11 Rovio bracht *Angry Birds* eerst uit bij een uitgever, die het spel op de markt zette en de publiciteit verzorgde. Het nadeel van een uitgever: je verdient minder aan je spel en bent afhankelijk van een derde partij voor marketing en plaatsing van het spel

Hoewel *Angry Birds* een belangrijke reden was voor EA om Chillingo te kopen, bleef dit spel eigendom van Rovio, dat inmiddels weer de rechten had. Het bleek uiteindelijk een goed idee om er niet helemaal afstand van te doen. Biedt een bedrijf je dus een jaarsalaris voor een origineel concept, dan is het goed om even op het idee te broeden. Wie weet neemt het spel zo'n vlucht dat het zelfs een beursgang mogelijk maakt.

1.7 Van ambitie naar gameontwikkelaar: een persoonlijke noot

In dit boek tref je een paar interviews met ontwikkelaars, waarvan een aantal vroeger met heel andere zaken bezig was. De interviews geven je een beeld van de markt en dienen antwoord te geven op de vraag: kan iedereen spellen maken en daar geld mee verdienen?

In 2007 had ik mailcontact met de eigenaar van Chillingo, het bedrijf dat erg bescheiden begon met het uitgeven van een klein aantal games waaronder *Angry Birds* (van Rovio) en *Zen Bound* (van Secret Exit). Ik deed wat tests en vertalingen voor *Zen Bound*, waar ik uiteindelijk voor werd bedankt in de credits (de aftiteling). Dat was niet alleen erg leuk, het wakkerde de passie voor games aan en vulde mijn hoofd meteen met allemaal ideeën.

In de jaren die volgden heb ik meer dan 1600 programma's gerecenseerd, waaronder honderden spellen. Ik had contact met talloze ontwikkelaars voor interviews, maar ook voor game-testing en het sporadische ontwerp van scripts of het maken van vertalingen.

Naast een vaste baan als softwaretester ben ik altijd betrokken gebleven bij het maken van games, bijvoorbeeld bij de grafische kant (het maken van menu's, knoppen, etc.) en het vertalen, proeflezen of schrijven van teksten.

Je leert een hoop tijdens het testen van software en als je dan ook nog software test waarmee je hele games maakt, dan gaat er een lampje branden. Zo is het idee voor dit boek ontstaan; ik wilde het gemakkelijkste programma belichten en uitleggen hoe je hiermee de eerste spellen maakt. Na uitvoerige tests met Gamesalad (voor de Mac), Game Maker (voor Windows en Mac) en Construct

2 (voor Windows) kwam ik tot de conclusie dat de laatstgenoemde de laagste instapdrempel had.



Figuur 1.12 Werk je mee aan een spel, dan is het leuk om een bedankje te zien in de aftiteling

Bovendien merkte ik dat Construct 2 constant wordt verbeterd, wat in mijn ogen belangrijk is voor de betrouwbaarheid van de software. Construct 2 is gemaakt door een kleine, frisse *startup* (een tweekoppig team uit Londen) en daardoor erg overzichtelijk. De makers koppelen op logische wijze functies aan gedrag en laten de gebruiker in kristalheldere schermen hun games ontwerpen.

Een belangrijke vraag blijft natuurlijk de volgende: hoe verdien je geld met een spel als je het eenmaal hebt bedacht én gemaakt? Voor het antwoord daarop heb ik diverse ontwikkelaars (groot en klein) geïnterviewd, alsmede een klein team dat een aantal jaren games heeft gemaakt, maar momenteel een kleine game-pauze heeft ingelast.

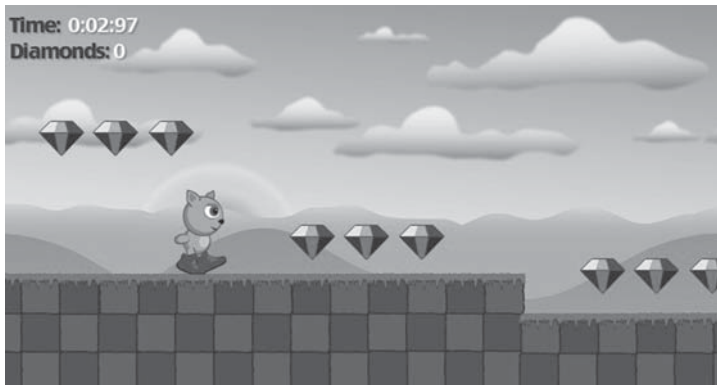
1.8 Interview: Launching Pad Games: succesvolle studio is even op 'break'

Zelf ben ik verzot op avonturenspellen. Ik speel ze al vanaf mijn jeugd en was dan ook blij dat er eindelijk een 'point and click'-game verscheen voor mijn iPhone. *Scarlett and the Spark of Life* was bedoeld als eerste deel van een reeks en bracht een prachtig verhaal naar mobiele telefoons. Het is gemaakt door een team van drie personen, dat eerder zelfs uit maar twee personen bestond.

Een aantal jaren geleden interviewde ik Tim Knauf en Tristan Clark, oprichters van Launching Pad Games, met betrekking tot het bovenstaande spel, dat een verrassend goede score kreeg op diverse gamesites.

Over Launching Pad Games

Launching Pad Games bestond aanvankelijk uit alleen de genoemde twee oprichters, met als eerste spel een simpele 'endless runner', ofwel een spel waarbij de hoofdpersoon automatisch loopt en jij moet drukken op een knop om te springen. Obstakels doemen op van tijd tot tijd en uiteraard moet je deze zien te ontwijken.



Figuur 1.13 *Run Boots Run* was de eerste game van het tweekoppige team van Launching Pad Games



Wil je meer weten over Launching Pad Games of een aantal van de spellen proberen? Ga dan naar www.launchingpadgames.com.

Na *Run Boots Run* maakte het team een prachtig puzzelspel in drie delen, dat je speelt vanuit de browser. Ga hiervoor naar www.launchingpadgames.com. Klik vervolgens op *The Pretender Trilogy* om het spel te spelen.

Tim Knauf en Tristan Clark maakten later *Zoo lasso*, een behendigheids spel voor de iPhone. Met de vergaarde kennis van mobiele spellen brachten ze

vervolgens *Scarlett and the Spark of Life* (deel 1) uit, dat niet onderdoet voor klassieke avonturenspellen in het ‘point and click’-genre.

Scarlett and the Spark of Life is helemaal gemaakt door Knauf en Clarke, maar na verloop van tijd besloot het tweetal een kunstenaar aan te trekken voor de grafische kant. Het heeft z'n vruchten afgeworpen, want het spel ziet er prachtig uit.



Figuur 1.14 Launching Pad Games heeft een aantal leuke games op haar naam staan. De ontwikkeling van hun spellen is vrij geleidelijk verlopen van het web naar mobiele apparaten

Voor deze basiscursus interviewde ik Tim Knauf, die met plezier een boekje opendeed over het opzetten van een gamestudio, het gebruik van verschillende programma's en het samenwerken in een team. Hieronder lees je zijn antwoorden.

Dit boek is gericht op iedereen die de droom koestert om eens een spel te produceren. Als ik (de gebruiker) een spel zou willen produceren, zou ik het dan helemaal alleen kunnen doen?

Zeker weten! Er is nooit eerder zo'n goede tijd geweest om je eigen games te produceren als nu. Er is ontzettend veel goed studiemateriaal beschikbaar – natuurlijk moet je daar wel even goed voor zoeken – en het internet maakt

het mogelijk om gelijkgestemden te vinden om kennis mee te delen. Dankzij de vele forums en communities is het maken van games en het bespreken van gebruikte technologieën een groot gemak.

Wat nog belangrijker is tegenwoordig is de mogelijkheid tot 'leren terwijl je creëert' met een grote keuze aan gemakkelijk te gebruiken gereedschappen en raamwerken; je hebt geen jaren aan studie meer nodig voordat je kunt beginnen. Dit is écht het gouden tijdperk voor games en laat niemand je anders vertellen.

Een van de hoofdstukken in dit boek is gericht op het vinden van goede middelen (kunst, plaatjes, etc.). Wat zie jij als absoluut minimum voor wat betreft de samenstelling van een team? Hoe is *Launching Pad Games* eigenlijk gevormd?

Je kunt echt wel mooie dingen produceren met een studio van 'slechts' één persoon. Ik had eerder individueel al een freeware (gratis) avontuurspel gemaakt, dat *Nathan's Second Chance* heette, voordat ik Tristan ontmoette. Tristan had eerder ook al een commercieel spel gemaakt en verkocht, dat *Fashion Star* heette. Hij werkte daarvoor (op afstand) met een artiest op contract. Het is dus echt wel mogelijk om een game te maken in je eentje, of vrijwel alleen en het is uitermate bevredigend om andere mensen jouw spel te zien spelen.

Dat gezegd hebbende, genoten we veel meer van het proces toen we begonnen met samenwerken. Een gemeenschappelijke vriend had ons op een feestje aan elkaar voorgesteld en we spraken daar over het ontwikkelen van spellen, tussen de slokjes cider door. Een aantal weken later waren we al met elkaar aan het werk voor een eerste spel (we hadden eerst trouwens heel andere plannen dan waar we uiteindelijk mee begonnen: *Run Boots Run*). Ik moet dus wel zeggen dat de minimumgrootte van een team om écht lol te krijgen in je werk – en eenzelvige mensen mogen het oneens zijn met me! – wel twee personen is.



Figuur 1.15 Scarlett is een prachtige *point and click*-, of eigenlijk *point and touch*-game voor iOS

De ontmoeting met onze artiest (de derde persoon in het team) was een andere leuke toevalligheid. Een ex-huisgenoot van een andere vriend, James, zocht werk op het moment dat Tristan ontevreden was met zijn eigen illustraties in ons iPhone-avontuur *Scarlett and the Spark of Life*. Gek genoeg hebben we toen alle kunst vervangen, maar dit was het waard: de stijl van James paste perfect bij het spel en inderdaad; hij paste zelf ook perfect bij Launching Pad Games.

Zou je het aanraden om altijd een gamedesign paper (een voorbereiding op papier) en extensieve prototyping (het testen van een spel met blokjes of simpele beelden) te gebruiken of is een idee voldoende om mee aan de slag te gaan? Is er bepaalde software die je kunt aanraden om prototypes mee te maken?

Documentatie van je ideeën is erg handig (en ik raad Google Docs of Basecamp aan als goede middelen voor samenwerking), maar niets is beter dan het maken van prototypes om de lol in een concept te vinden. Het hoeft niet geavanceerd te zijn: afhankelijk van het type spel kun je zelfs met papier een paar ontwerpen maken. Voor een actiespel werkt het niet zo goed, dat spreekt voor zich, het is dus goed om tijd te investeren in het vinden van een goede digitale gereedschapskist waarmee je kunt proefdraaien wanneer inspiratie de kop opsteekt.



Figuur 1.16 Cocos2D is een engine, ofwel motor, voor het maken van iPhone-spellen in 2D. In hoofdstuk 3 lees je meer over engines

Veel mensen zweren bij Adobe Flash voor het uitproberen van ideeën, hoewel ik het persoonlijk te veel friemel- en frunnikwerk vind. Tristan heeft zo goed leren werken met Multimedia Fusion (een totaalpakket voor gamedesign) dat hij zijn creaties moeiteloos door brandende hoepels laat springen – letterlijk zelfs.

James heeft wel eens gespeeld met GameSalad (een complete gamestudio voor de Mac). Persoonlijk werk ik het liefste met de gereedschappen die me het beste zijn bijgebleven (wat in het geval van onze spellen voor de iPhone altijd Cocos2D is geweest). Het belangrijkste is dat je een voor jou vertrouwd pakket gebruikt, zodat je snel kunt werken en niet over technische toestanden hoeft na te denken.

Dit boek is gericht op het werken met Construct 2, zodat er geen drempels zijn die te maken hebben met programmeertalen of kennis daarvan. Wat was voor jullie de keuze op het moment dat je het eerste spel maakte? Had je achteraf gezien gekozen voor andere software?

Ons eerste spel was *Run Boots Run*, dat helemaal gemaakt was in Flash, met Actionscript 3 (software van Adobe, waarmee je animaties en spellen kunt maken). Ons eerste spel voor de iPhone was *Zoo Lasso*, waar we Cocos2D voor gebruikten. Op dat moment leek het alsof we in beide gevallen niet veel keuze hadden – engines als Unity of HTML5 kwamen nét van de grond – en buiten Cocos2D om waren er nog geen totaalpakketten voor het maken van games.

Zou ik opnieuw moeten beginnen vandaag de dag, dan zou ik zeker gaan voor een gemakkelijke oplossing met HTML5-technologie. En ja, ik zou zeker eens gaan werken met Construct 2!

Run Boots Run en The Pretender Trilogy zijn webgames, Zoo Lasso was jullie eerste iPhone-game. Hoe verliep deze overgang? Inhakend daarop, wat zorgde ervoor dat je de overstap naar iOS maakte?

Run Boots Run was echt een leerproces, maar we werden serieus toen we begonnen met *The Pretender Trilogy*. We lanceerden het spel op: www.flashgamelicense.com (tegenwoordig www.fgl.com) en sloten een overeenkomst met Spil, een uitstekende uitgever van Flash-games. Dat was goed voor de bankrekening van het bedrijf, maar het was een eenmalige betaling.

Zouden we het bedrijf levensvatbaar willen houden op de lange termijn, iets dat nooit echt is gelukt (maar we hebben het geprobeerd!), dan moesten we een constante geldstroom zien te bewerkstelligen. De App Store van Apple was net helemaal hot aan het worden en het voelde als een goede mogelijkheid om mee te doen voordat grote ontwikkelaars de boel zouden overnemen, iets dat eigenlijk volledig is gebeurd in de jaren die volgden.

De overgang van web naar iOS ging aardig gesmeerd. We gebruikten nog steeds dezelfde programma's (Flash voor het tekenwerk, Ableton Live voor de muziek) en Cocos2D was ook een goede keuze. Bovendien is Cocos2D voorzien van goede handleidingen, een mooi design en een sterke online gemeenschap. Op het moment dat wij ermee werkten was de focus nog slechts op iPhone-games, waarna het al snel complexer werd.

Als je een eerste spel om te maken zou aanbevelen, wat betreft een genre of schaal, wat zou het zijn?

Wat betreft schaal in ieder geval: klein, klein en nog eens klein! Je wilt het vertrouwen in jezelf als ontwikkelaar opbouwen en als je idee veel afwerking en/of technische kennis vereist of een grote omvang heeft, dan krijg je het zwaar en kun je gedemotiveerd raken (terugkijken naar mijn avonturenspeel *Nathan's Second Chance* was dit een slechte keuze voor mij als eerste project).

Uiteindelijk kostte dit project me drie jaren van mijn vrije tijd. Dus ja, laat die plannen voor het maken van een nieuwe *Zelda*- of *Grand Theft Auto*-achtige game maar even liggen tijdens je eerste (en tweede) project. Kies een genre dat je zelf interessant lijkt. Veel mensen raden aan om een *Tetris*-kloon te produceren als eerste spel, maar hoezeer ik ook van *Tetris* houd, het spel klonen zou me mateloos vervelen. Ga vooral je gang en pas dat originele idee van je toe, maar let erop dat er genres zijn die automatisch een enorme omvang krijgen: actie-adventures, strategie en natuurlijk role playing games. Puzzelspellen, platformers, 2D-shooters en bordspellen zijn veel beter te bolwerken.

Register

Symbolen

3D-game 39
8 directions 54, 81
8-directions, gedrag 145

A

aanraakbesturing 142
aanraakscherm 140
achtergrond 47
 Pixlr.com 74
 toevoegen 49
Adobe Flash 28
Adult Swim Games 136
Angry Birds 125
App stores 137
Asteroids 94
attributes 52
Audacity 35

B

Behaviors 20, 52
besturingsmethode 140
beweging 51, 54
botsingen 66
Breakout 72
Bullet 20, 40

C

cheat sheet
 Asteroids 110
 Breakout 93
 Pong 71
Cocos2D 28
Construct 2 19, 37
 exporteren uit 135

D

draaien van object 74
Dream Heights 13

E

eigenschappen 52
eindscherm 91, 110
endless runners 140
engine 38
Events 42
Event Sheet 20, 45
exporteren 41
 vanuit Construct 2 135

F

Fade 39
Fez 128
Flash Game License 136
freemium 9
freesound.org 35

G

Game maker 120
GameSalad 28
gamesites 136
geluid 147
 gratis 35
General 53
global variable 46, 67
grafische bewerker 43
gratis geluid 35
gratis materiaal 34

H

heads-up display (HUD) 84
Home 42

I

idee ontwikkelen 12
 importeren van object 60
 Input, categorie 140
 instance variable 80

J

Jump-thru 52

K

kaatsbeweging 63
 keyboardbesturing 144
 kleurverloop 48
 knoppen op het scherm 142
 Kongregate 136

L

lagen 50, 75
 Launching Pad Games 25
 Layout 20, 42
 levens
 toevoegen 81
 verliezen 84
 lostgarden.com 34, 114

M

movements 54
 Move to layer 144
 muisbesturing 145
 Multimedia Fusion 28

O

object
 draaien 74
 gedrag koppelen 102
 gedrag toewijzen 80
 importeren 60, 98
 maken 52
 Object bar 44
 On start of layout 64
 Opacity 143
 outsourcen 34

P

Phoneyap 41
 Pixel'd 17
 pixel-editor 14, 97

pixels 15
 Pixie-engine 16, 97
 pixlr 47
 plagiaat 13, 31
 Platform 20, 40
 platformspel 113
 Pong 58
 poortwachter 44
 Preview on LAN 145
 Properties 52
 prototype 28

R

Rectangle 52

S

scherm lay-out 42
 schermresolutie 40
 score toevoegen 67, 81, 86
 Smurfs' Village 10
 Snap to grid 78
 Solid 52
 spelregels toevoegen 81
 sportspellen 124
 Sprite Maker 18
 sprites tekenen en opslaan 76
 stuiterbeweging 65, 66
 Super Meat Boy 128
 symmetrisch potlood 16
 System Event 64

T

tekstobject 67, 102
 testen 91
 Tiny Tower 13
 Tiny Wings 32
 top-down shooter 114
 Touch 139
 Triple Town 13

V

verloopgereedschap 48
 View 42
 voorgrond 50

W

Where's My Water 125

winnen 68, 90

World of Goo 127

Wrap 102

BASISCURSUS GAMEDESIGN

Wil je een eigen game maken voor een website, smartphone of tablet? Een goed idee is belangrijk maar waar begin je? Hoe maak je een eerste proefversie? Hoe zet je jouw game vervolgens online of presenteert je het bij een digitale winkel met apps en games?

In deze *Basis cursus* leer je vanuit heldere hoofdstukken hoe je een game ontwerpt, van het eerste idee tot publicatie en alles er tussenin. De auteur geeft antwoord op alle praktische vragen. Wat zijn de juiste oplossingen voor de obstakels waar je tegenaan loopt? Hoe kom je aan een ontwerp en onderdelen als kunst en geluid voor je game?

Het boek is echter niet alleen praktisch van aard. Speciaal voor deze *Basis cursus* zijn een aantal succesvolle gameontwikkelaars geïnterviewd. Laat je inspireren door de talloze tips die zij geven en haal het maximale uit jouw game!

Met dit boek leer je games maken in Construct 2. Werken met deze gratis software geeft programmeren een hele andere betekenis. Vergeet code, formules en schermen met alleen maar cijfers en teksten: Construct 2 maakt van al deze code een visuele weergave die voor iedereen te begrijpen is. Dit maakt deze *Basis cursus* bijzonder geschikt voor beginnende gameontwikkelaars, maar ook voor ervaren designers. Construct 2 versnelt namelijk het ontwikkelproces en haalt de noodzaak voor het invoeren van ellenlange regels code weg.

OVER DE AUTEUR

Merijn de Boer is auteur en recensent bij Europa's grootste downloadportal voor software & games. Zijn recensies en boeken delen hetzelfde doel: informatie toegankelijk maken voor een groot publiek.

ISBN 978 90 12 58538 5

NUR 980



9 789012 585385

www.academic-service.nl