

● Games maken (2) ●

Het vervolg op het artikel in SoftwareBus 2021-5 over het ontwerpen van spellen, oftewel Game Design.

René Suiker

Belofte maakt schuld

Natuurlijk was het kort dag tussen het uitkomen van de vierde editie en het schrijven voor de vijfde editie van dit jaar, maar ik had het e-mailadres GameOntwerp@CompUsers.nl nog niet in mijn mailclient geïnstalleerd.

Nu is het voordeel van die korte periode natuurlijk dat het nog niet storm loopt op dit adres, maar intussen heb ik de eerste aanmelder al van een reactie kunnen voorzien. We kunnen dus nog mensen gebruiken om dit initiatief mee vorm te geven en het e-mailadres is operationeel.

Terugblik

In de vorige aflevering hebben we Unity® geïnstalleerd en we zijn eens naar het openingsscherm gaan kijken. En mogelijk zijn we allemaal overdonderd door de hoeveelheid instellingen die je kunt maken. Hier is de kans groot dat je door de bomen het bos niet meer ziet, maar we gaan de komende tijd proberen de weg te vinden.

Met Scratch ben ik inmiddels (zie in het vorige nummer) op artikel 14 gekomen, met Unity moeten we dit aantal toch ook wel zien te halen. Intussen heb ik in elk geval groen licht om op te starten, we moeten nog zien hoe levensvatbaar e.e.a. is, maar het kan toch op zijn minst een interessante serie artikelen opleveren. Maar nogmaals, het zou ook leuk zijn als anderen kunnen bijdragen.

Game-industrie

We hebben ons als CompUsers altijd met serieuze zaken bezig gehouden, maar onderschat intussen de game-industrie niet. Daar gaat intussen al meer geld in om dan in de film-industrie. En als je een game van enige omvang wilt maken, dan heb je daar een aantal disciplines bij nodig.

In een ander artikel komt aan de orde dat je met BASIC of met Scratch wel een game kunt maken, maar daarmee ga je geen prijzen winnen. Als je echt indruk wilt maken en wat verkopen wil boeken, dan moet je echt wel iets te bieden hebben. En als het even kan, dan komen in ons 'project' al deze aspecten aan de orde.

Wat heb je dan zoal nodig:

1. Bedenker, in feite de game-designer
2. Grafische artiesten
3. Muziek- en geluids-ontwerpers
4. Projectmanager
5. Programmeurs
6. Marketingdeskundige(n)

Misschien vergeet ik ook nog een en ander, maar dit heb je minimaal nodig om een succesvol spel in de markt te zetten. Een spel is dus ter ontspanning van degene die het speelt, maar het is een heel serieuze business om een goed spel te maken. Daarnaast heb je dan ook nog de zogenaamde 'serious games', dat zijn dus niet zomaar spellen ter ontspanning,

maar een speelse wijze om een bepaald doel te bereiken. Vaak zijn die voor een heel selecte doelgroep geschreven.

Programmeren of niet

Vooralsnog gaan we ervan uit dat we met Unity als game-engine gaan werken. Zoals ik eerder al schreef, de grens tussen programmeren en applicaties begint te vervagen. Daarom hebben we tegenwoordig ook wat aandacht voor programmeren en als je zelf spellen maakt, dan ben je dus wel aan het programmeren. Maar het merendeel van wat je in een game-engine doet is configureren. En dat doe je in veel serieuze applicaties ook.

Als je kijkt wat we bijvoorbeeld in Corel Draw allemaal in konden stellen op een afbeelding, dan begon dat ook al op programmeren te lijken. En als je dan in een workflow wat bewerkingen begint te automatiseren, dan lijkt het nog meer op programmeren.

Kortom, programmeren is geen vies woord meer binnen CompUsers. En ik ben daar bepaald niet verdrietig om; laten we niet vergeten: in de begin dagen, toen we nog DOSgg heetten, hebben we toch ons eigen boekhoudprogramma (HABO) geschreven en uitgebracht.

HCC!programmeren

Even een klein zijstapje. Hcc!programmeren is ook een interessegroep binnen HCC en zij ondersteunt iedereen die zelf software wil maken. Het is ook een actieve ig, die bijna altijd ook op onze evenementen te vinden is.

Net als dat wij binnen CompUsers Platforms hebben, hebben zij ook specifieke gespecialiseerde werkgroepen met eigen activiteiten. Vorige keer maakten we al kennis met Gordon, die erg actief is in deze ig, en die zich erg richt op programmeren in Liberty Basic. De ig heeft net als wij een eigen website (<https://programmeren.hcc.nl/>); zij verzorgen per interesse regelmatig terugkerende sessies, zowel online als in De Bilt. Voor deze artikelenreeks werken we intensief samen met hun werkgroep 'games programmeren'.

Plannen

We zijn dus van plan om, in samenwerking met in het algemeen HCC!programmeren, en speciaal met Marco Kurvers, een reeks van artikelen op te zetten, zodat we eventuele liefhebbers in onze vereniging goed op weg kunnen helpen om zelf games te maken.

Want wat is er nu leuker dan je (klein)kind een spel te laten spelen dat je zelf gemaakt hebt. Ik weet nog hoe mijn oudste zoon ooit voor Sinterklaas een game had gemaakt als surprise voor mijn jongste zoon, die door bepaalde levels uit te spelen hints kreeg over waar het echte cadeau verstopt was. Iedereen was onder de indruk van dit idee. Dat was gemaakt met GameMaker; daar ga ik misschien ooit nog eens wat in wroeten, maar nu eerst verder met Unity.

We hebben een opzet waarin we komend jaar gaan uitleggen hoe je een 2D-game in elkaar zet; in 2023 kunnen we ons dan verdiepen in 3D-games. Genoeg te doen, dus laten we maar beginnen.

Toepassing van Unity

Unity is dus vooral een game-engine. Een engine is letterlijk te vertalen als ‘motor’ en een game-engine is in feite de motor achter een computerspel, dus een stuk basissoftware, die het maken van een computerspel vergemakkelijkt. Veel zaken die je in bijvoorbeeld Basic of Java allemaal met de hand moet programmeren, kun je in zo’n game-engine configureren en dan zorgt de engine er voor dat het spel zich zo gedraagt als je wilt.

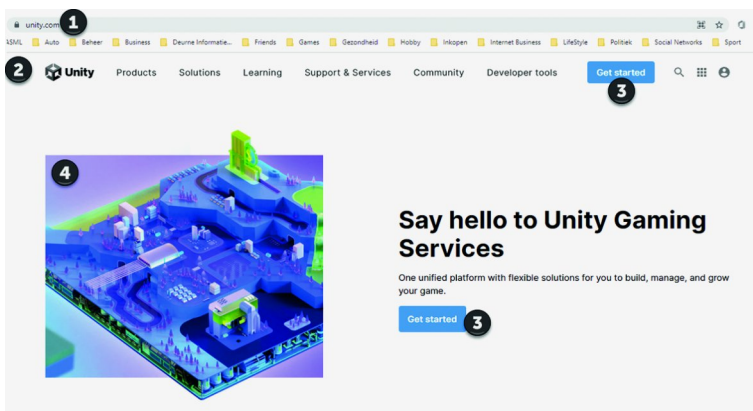
Uiteindelijk zul je voor een goed spel ook moeten programmeren en in combinatie met Unity doe je dat in principe in C#. Unity heeft een zeer goede integratie met Visual Studio van Microsoft, waarin je vrij eenvoudig met C# aan de slag kunt. Dit is allemaal erg theoretisch en misschien nog wat moeilijk te volgen, maar we gaan het straks in de praktijk brengen en dan gaat dit hopelijk vanzelf een beetje leven.

Naast gebruik als game-engine zijn er ook andere toepassingen van Unity, o.a. voor het aansturen van 3D-printers en ander computergestuurd gereedschap. Misschien dat we daar ooit nog eens aan toekomen, maar vooralsnog richten we ons voornamelijk op het zelf maken van computerspellen. Verder zeggen ze (van Unity zelf) ook dat het geschikt is voor andere interactieve ervaringen.

Vorige keer hebben we kort gesproken over de installatie van Unity en heb ik bepaalde aspecten van het scherm uitgelegd. Vandaag gaan we iets dieper in op beide aspecten en we gaan een stukje met de regels van de natuurkunde spelen. In de loop van deze reeks gaan we echt programmeren, maar vandaag beperken we ons tot Unity zelf. En we nodigen je uit om mee te doen op je eigen computer. Om een beetje een idee te krijgen van wat je met Unity kunt: <https://unity.com/madewith>.

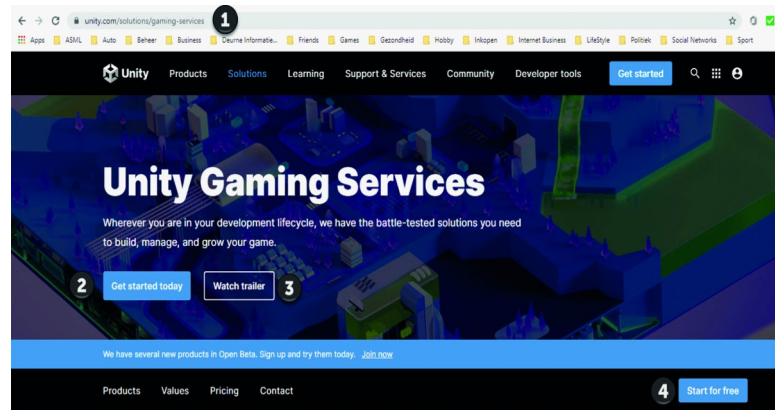
Installatie

Zoals ik al aangaf gaan we deze keer iets uitgebreider in op de installatie. We gaan naar <https://unity.com> om Unity te downloaden, zodat we het kunnen installeren. We komen dan op de site zoals hieronder aangegeven:



Afb 1 De site van Unity

Bij (1) zie je de URL, zoals aangegeven. Bij (2) zie je de menubalk van de site. Je hebt hier verschillende opties, maar we klikken op optie (3), ‘Get started’. De (4) is ingevuld, om even de aandacht te richten op dit mooie scenario, dat ongetwijfeld met Unity is gebouwd. We komen dan op de pagina hieronder:



Afb. 2 Het starten

Hier zie je ook weer de URL bij (1), om te laten zien dat we nog steeds een site bezoeken, we zitten nog niet in de applicatie. Je kunt hier gelijk starten (2), maar je kunt ook nog een trailer bekijken (3). Bij (4) gebeurt hetzelfde als bij (2), maar het vestigt nog even de aandacht op het feit dat je gratis met Unity kunt starten.

Uiteraard willen de ontwikkelaars van Unity ook geld verdienen, maar dat gaan ze pas doen bij uitgebreidere versies van Unity of wanneer je als klant intussen veel geld verdient met de spellen die je maakt op basis van Unity. Zolang je nog onder de ton aan omzet blijft hoeft je voor het gebruik van Unity niet te betalen.

En zodra je veel geld gaat verdienen met de verkoop van je spellen, is het natuurlijk ook reëel dat je een redelijke vergoeding betaalt voor de engine die je zoveel werk uit handen neemt.

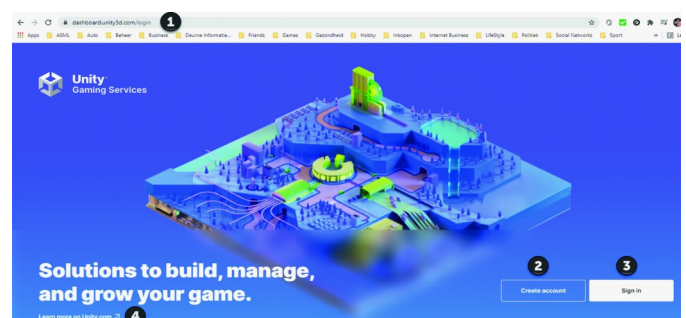
Het is natuurlijk leuk om even naar de trailer te kijken, je wordt dan doorgestuurd naar YouTube: https://www.youtube.com/watch?v=ac_N59a7_Os voor een filmpje over de mogelijkheden van Unity. Als je daar toch gaat kijken, kun je je natuurlijk gelijk abonneren op het YouTube-kanaal van Unity. Daar is namelijk nog veel meer te vinden wat je later op weg kan helpen. Maar dit terzijde, we gaan Unity installeren. Ze stellen zelf dat je in drie stappen aan de slag bent en dat is waar, maar later blijkt dat er nog wel iets meer nodig is, maar daar komen we tegen die tijd dan wel op terug. De stappen zijn:

1. Download en installeer Unity Hub
2. Kies de Unity-versie waarmee je wilt werken
3. Start je project

En op zich klopt dat ook. We nemen het stap voor stap even door.

Unity Hub

De Unity Hub is in feite een soort voorportaal van waaruit we onze activiteiten met Unity opstarten. Ik ga ‘m straks wat uitgebreider beschrijven, maar vooralsnog is het goed om te weten dat hier inderdaad onze reis door Unity begint. Als we op de knop (2) hiervoor gedrukt hebben, komt het volgende scherm tevoorschijn:



Afb. 3 Downloaden

Bij (1) weer de URL, bij (2) de knop om een account aan te maken, bij (3) de knop waarmee je verder kunt als je al een account hebt. We gaan straks wel voor de gratis versie, maar we gaan wel een account aanmaken. Let op, bij (4) een link om nog veel meer over Unity te leren. Klik op (2) en het volgende scherm verschijnt:

Afb. 4 Account aanmaken

Bij (1) geef je je e-mailadres in, bij (2) een gebruikersnaam die je wilt gebruiken, bij (3) een wachtwoord dat je daarbij gaat gebruiken, bij (4) een aantal afspraken die je moet of kunt accepteren en bij (5) geef je aan dat je geen robot bent. Bij (6) kun je je account aanmaken, je kunt er ook voor kiezen om via een social media-account een Unity-account aan te maken (7).

Heb je alles naar behoren ingevuld, en heb je aangetoond door middel van het selecteren van wat vakjes op een foto dat je toch wel iets menselijks hebt, dan krijg je de volgende bevestiging:

Afb. 5 Bevestiging e-mail

We kijken dus even in de mail en daar treffen we inderdaad iets aan waar we op moeten reageren:

Afb. 6 Bevestiging in de e-mail

Ik klik dus op de link in de e-mail, bevestig daar nogmaals dat ik geen robot ben en moet dan via mijn e-mailadres en het wachtwoord inloggen. Geen idee wat ik met die user-

name moet doen, maar die blijft in elk geval opgeslagen. Vervolgens keer ik terug naar het scherm van afb. 5 en klik op (2) 'continue'. Ik kom dan op de Unity-site op de pagina van mijn account.

Afb. 7 Mijn account

Hier zie je bij (1) weer de URL waar we zijn terechtgekomen, bij (2) zie je dat je ingelogd bent; als je klikt op het pijltje rechts ervan dan kun je bij (3) Unity downloaden. En dat gaan we doen. We komen dan op het volgende scherm, waar we iets uitgebreider bij stilstaan:

Afb. 8 Unity downloaden

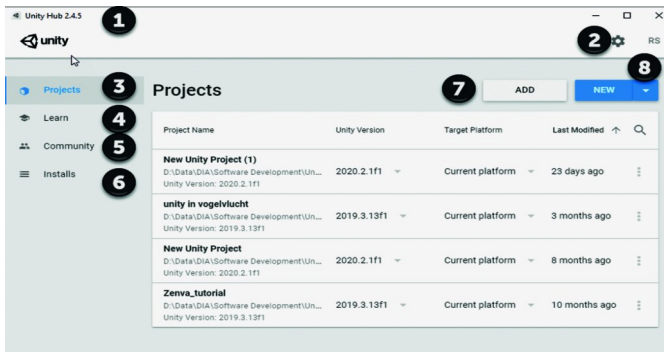
Bij (1) weer de URL, maar die gaan we nu niet meer heel veel tegenkomen. Bij (2) de knop om de Hub te downloaden en dat gaan we zo doen. Bij (3) staat een opmerking om de Unity versie 2021.2 te gaan downloaden, maar dat gaan we straks via de Hub doen, dus niet via dit scherm.

Bij (4) de systeemeisen om de Hub te kunnen installeren. Unity is niet alleen voor Windows verkrijgbaar (maar die versie gebruik ik wel bij het schrijven van dit artikel), maar ook voor MacOS en bijvoorbeeld Ubuntu.

Bij (5) wat opmerkingen over releases en speciale versies die wat langer ondersteund blijven. Ten slotte bij (6) nog wat links naar diverse relevante pagina's, waarbij ik met name die naar 'Documentation' zeker kan aanbevelen, want er is enorm veel documentatie voorhanden. Elk aspect van Unity is gedocumenteerd. In deze artikelenreeks ga ik natuurlijk heel veel uitleggen over veel aspecten van Unity, maar ten opzichte van deze documentatie blijf ik echt aan de oppervlakte; daar kun je echt de diepte in als je details zoekt over bijvoorbeeld de Hub, maar ook over game-objecten.

Als je dus op (2) klikt, dan kun je de software downloaden, en ik ga er vanuit dat jullie dat allemaal zelf kunnen. Tot zover dus de uitleg over het downloaden en installeren van de Hub. Eenmaal geïnstalleerd, start je de Hub op en die zie je hierna in afb. 9.

Bij (1) geen URL meer, we zitten nu in een programma, niet meer in de browser. Vanuit de Unity Hub starten we Unity op, maar dan moeten we wel de Unity-editor geïnstalleerd hebben. Dat gaan we zo doen, maar eerst nog even wat aandacht voor de Hub.



Afb. 9 De Unity Hub

Bij (2) zie je een knopje voor de instellingen, ofwel de 'preferences'. Niet heel spannend voor dit moment. Ook een knopje over je account; op dit moment ook nog niet heel spannend.

Bij (3) ga je naar de projectenpagina. Hier start je je projecten op of open je bestaande projecten. De knoppen (7) en (8) zijn ook aan deze pagina gekoppeld, ze laten je toe een nieuw project te starten (8), op basis van beschikbare templates.

Bij (7) krijg je de optie om een project dat je ergens op een schijf hebt staan ook binnen jouw Unity-versie te laden. Bij (4) zijn wat opties om te leren. Dit is vooral voor studenten bedoeld, maar je kunt er misschien nog wel wat van opsteken. Voor deze artikelen laten we het even voor wat het is.

Bij (5) krijg je wat zaken over de 'Community'. Dit is vooral bedoeld voor professionele gamemakers. Dat zijn we voorlopig nog niet, dus dat laat ik ook nog even links liggen.

Bij (6), tenslotte, komen we op de Unity-installaties en daar gaan we weer even wat dieper op in.

Unity-versie

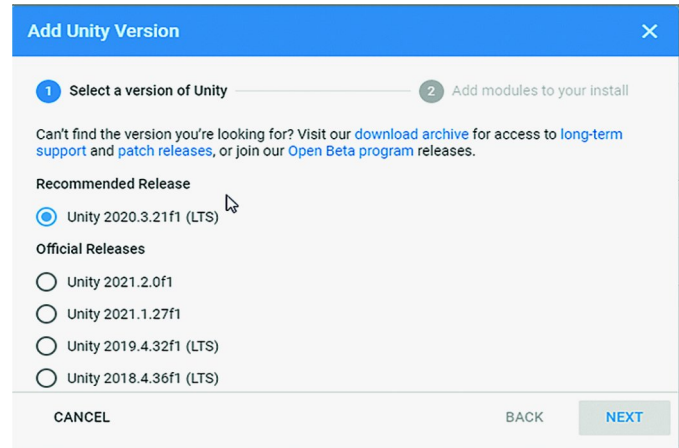
Weet je het nog, de drie stappen van zojuist: de Hub hebben we geïnstalleerd en nu gaan we dus naar de Unity-versies kijken. En als we dat gedaan hebben gaan we een project opstarten, dus komen we weer in de Hub en gaan dan weer naar de projectenpagina. Maar eerst dus een versie van Unity installeren, anders kunnen we nog helemaal niets. Klik dus op (6) voor de installaties en het volgende scherm komt op:



Afb. 10 Unity-installaties

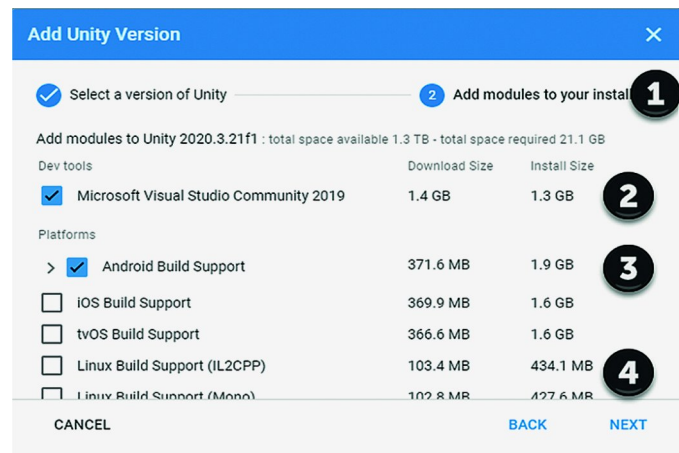
Ik heb natuurlijk al even wat geëxperimenteerd voordat ik deze artikelenreeks begon; ik ben nog niet zo heel ver wat mijn kennis betreft, maar ik heb wel diverse Unity-versies staan. Die zijn allemaal niet zo heel groot (iets meer dan 1 GB, dus groot is ook een relatief begrip) en je kunt er gerust meer dan een laten staan.

Maar ik zou ook best eens opruiming kunnen houden als ik diskruimte te kort ga komen. Meestal koop ik er dan gewoon een disk bij, maar ja, dat is ook geen oneindige oplossing. We zaten dus bij (1), de installaties. Onder (2) zie je welke installaties je hebt en ik veronderstel dat, als je nu begint met Unity, er nog niet één staat. Klik daarom bij (3) op 'Add' om er een (of meer) toe te voegen:



Afb. 11 Versies van Unity

Je ziet hier een aanbevolen versie bovenin staan, daaronder wat officiële versies. Scroll je naar beneden, dan zie je ook nog een pre-release-versie staan. Wij klikken gewoon op 'Next' want we willen met de aanbevolen versie aan de slag. Daarna kiezen we nog enkele modules:



Afb. 12 Aanvullende modules

Je ziet bij (2) dat je nu de additionele modules kunt toevoegen aan deze installatie. Bij (2) heb ik de Microsoft Visual Studio aangevinkt. Visual Studio is goed geïntegreerd met Unity, en omdat we later gebruik gaan maken van C#-scripts is het handig om deze alvast aan te vinken.

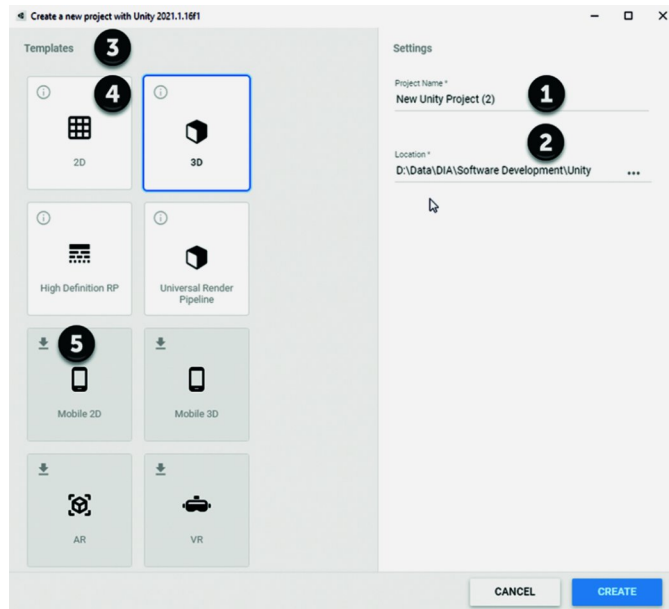
Bij (3) vind je de Platforms waarvoor je kunt bouwen. Ik heb hier de Windows-versie geïnstalleerd, maar kan desondanks ook voor Android (geselecteerd) bouwen en bijvoorbeeld ook voor Linux. Hieronder staat dat je ook nog de volledige documentatie kunt downloaden. Als je altijd toegang tot Internet hebt is dat een wat overbodige optie, want de hele documentatie is online voorhanden. Maar als je voldoende ruimte op je harde schijf hebt, waarom niet. Klik bij (4) op 'Next'.

Je moet dan nog wel de licentievoorzwaarden voor Visual Studio accepteren en dan druk je weer op Next. Omdat ik voor Android heb aangevinkt, moet ik ook hier de Android-voorwaarden accepteren en dan klik ik op 'Done'. Overigens, ik heb nog nooit iets voor Android gemaakt, althans niet via Unity, maar het schijnt dat je er nog uitbreidingen voor moet downloaden. Dit komt misschien later in de reeks nog eens aan de orde, maar dan ook echt later, niet in 2022.

Vervolgens wordt de geselecteerde versie (met de aanvullende modules) op je pc geïnstalleerd en afhankelijk van je Internetverbinding kan dat een paar minuten tot enige uren duren. Als de versie geïnstalleerd is, is het scherm onder 'Installs' (zie afb. 10 bij (2)) niet meer leeg, maar zie je de zojuist geïnstalleerde versie staan.

Je project

We gaan dan weer naar 'Projects' voor de derde stap in het geheel en kiezen 'New' om een nieuw project te starten:



Afb. 13 Nieuw project

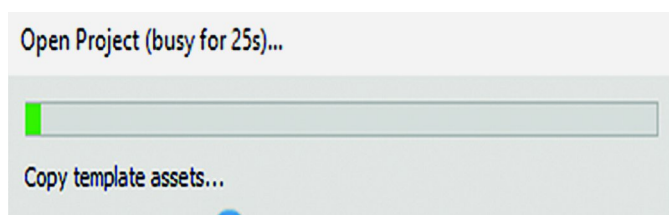
Bij (1) geef je een naam aan je project. Je kunt wel de naam laten staan die Unity gegeven heeft, maar die dekt zelden tot nooit de lading. Voor dit artikel start ik met een project dat ik 'Basis mechanica' noem.

Bij (2) vul je de locatie in waar je je project wil opslaan. Je zag net eerder een knopje voor locaties dat ik oversloeg, maar je kunt voor Unity ook een standaardlocatie voor je projecten instellen.

Bij (3) kies je welke template je wilt gebruiken. Voorlopig beginnen we de reeks met 2D-uitleg, in 2023 kunnen we eens naar 3D gaan kijken: we kunnen niet alles tegelijk. En uiteraard onder het voorbehoud, dat de artikelenreeks doorgaat, want als er geen belangstelling is dan houdt het een keer op. Maar vooralsnog ga ik ervan uit dat hier ook belangstelling is en dat we net als met Scratch op een echt feuilleton kunnen rekenen.

Bij (5) zie je dat je nog wat moet downloaden wil je een spel voor je GSM maken. Zoals gezegd, voorlopig nog even niet.

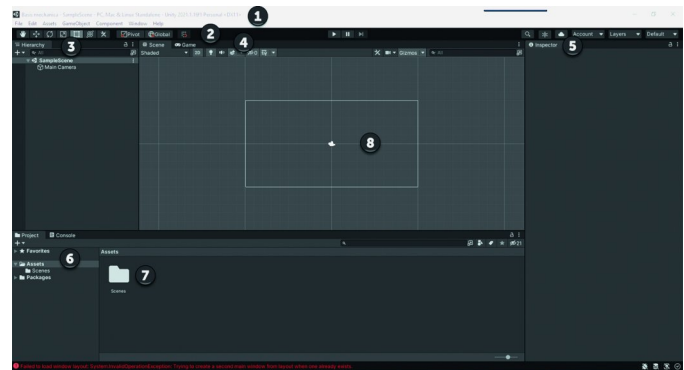
Klik dus op (4) en vervolgens op 'Create', rechts onderin. Nu wordt het project aangemaakt: vanuit Unity wordt een en ander voorbereid. Dit kan even duren:



Afb. 14 Project aanmaken

De editor

Als dat eenmaal gebeurd is, start de editor op: we kijken even naar de editor als geheel en zoomen later in op een aantal details.



Afb. 15 De Unity-editor

Bij (1) zie je de titelbalk met de naam van het project, de scene en de versie van Unity die we gebruiken. Ook zie je nog 'personal' staan, waarschijnlijk een gevolg van het feit dat ik een personal licentie heb. Het stuk over licenties en account is in feite nauwelijks aan de orde gekomen, waarschijnlijk omdat ik dit allemaal al geregeld had en ik dus de pop-up niet kreeg. Maar als daarom gevraagd wordt: we maken een personal account aan met een personal licentie. Die is gratis, zolang je nog onder de 100 ton omzet blijft. Ik hoop van harte dat we allemaal voor onze licenties gaan betalen, want dat betekent dat we onze spellen kunnen verkopen.

Onder de titel zie je een menubalk, zoals je in veel applicaties tegenkomt. Hier staan natuurlijk wat keuzes in die je niet in alle programma's tegenkomt, maar daar gaan we in een later stadium wat dieper op in.

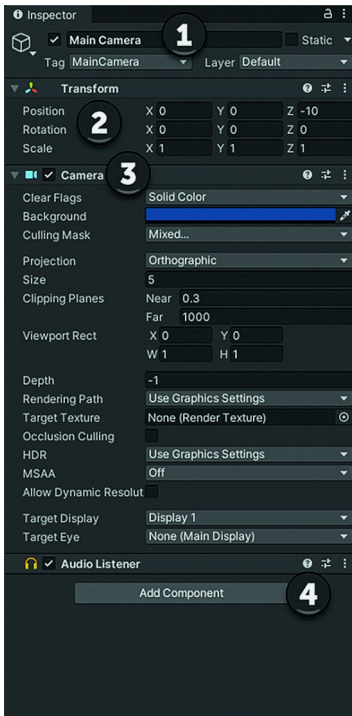
Bij (2) kom je op een knoppenbalk (toolbar) met opdrachten die je gedurende de spelopbouw kunt gebruiken. Bij (3) zie je de hiërarchie: alle spelobjecten (game objects) in tekstvorm. Bij (4) zie je de scène, waarin je je spel opbouwt. Bij het aanmaken van het project wordt er altijd minimaal één scene aangemaakt en is er altijd een camera als object aangemaakt. De inhoud van de scène en de hiërarchie zijn aan elkaar gekoppeld; ze tonen dezelfde items in een andere vorm, maar als je iets selecteert in bijvoorbeeld (3), dan is het ook in (4) geselecteerd en vice versa.

Bij (5) zie je de 'inspector'. Hier zie je allemaal parameters die je voor het geselecteerde object kunt invullen. Alles wat je in je spel gebruikt zijn game-objecten. Deze objecten kunnen weer meerdere componenten bevatten om de werking van zo'n object te beïnvloeden. Dit klinkt allemaal heel theoretisch, maar we gaan dit straks in de praktijk brengen en in de loop van de tijd gaat dit duidelijk worden.

Bij (6) zie je de assets en ook de favorieten. De favorieten laat ik nog even voor wat het is. Bij assets zie je alles wat je in je spel kunt gebruiken. Als je het in de hiërarchie opneemt, dan zit het in je scene(s), maar soms weet je al dat je dingen wilt gebruiken en dan organiseer je dat alvast binnen je assets en dan kan je ze later naar je scene halen. Ook hier komen we later uitgebreid op terug.

Bij (7) een overzicht van de inhoud van de folder die je bij (6) hebt geselecteerd. Je ziet in het voorbeeld dat we maar één asset hebben; dat gaat nog veranderen in de loop van de opbouw, maar deze kregen we al mee met het aanmaken van het project: scenes. Binnen Unity speelt alles zich af binnen een scene. Een spel kan meerdere scenes bevatten, maar heeft er altijd tenminste één. Als je dubbelklikt op de folder 'Scenes' in (7) of je selecteert 'Scenes' onder (6), dan kun je zien dat er één scene bestaat (SampleScene).

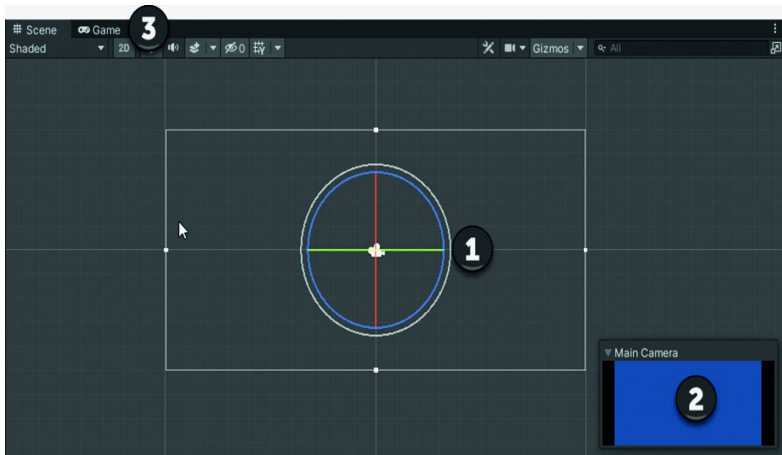
Bij (8) ten slotte zien we de scene met de objecten die erin geplaatst zijn. In dit voorbeeld alleen de camera. Klik je op de camera, dan zie je dat er parameters voor de camera bij de inspector (5) zichtbaar worden gemaakt. Laten we eens in detail kijken naar de inspector:



Afb. 16 De inspector in detail

dus specifieke zaken inzake de camera.

Bij (4) zie je de mogelijkheid om componenten aan dit object toe te voegen. Dat gaan we nog niet doen bij de camera, maar straks wel bij andere objecten. Als je nu trouwens goed kijkt, terwijl de camera is geselecteerd binnen de scene (of de hierarchy, dat werkt net zo goed), dan zie je binnen de scene een extra blokje in beeld; dat toont het blikveld van de camera:



Afb. 17 De scene

Bij (1) kun je aan de blauwe cirkel zien dat dit object geselecteerd is. Tevens zie je de X-as en de Y-as afgebeeld.

Bij (2) zie je het beeld dat de camera ziet. Dat is ook het beeld dat je ziet als je bij (3) kiest voor 'Game' in plaats van 'Scene', maar dan wat kleiner.

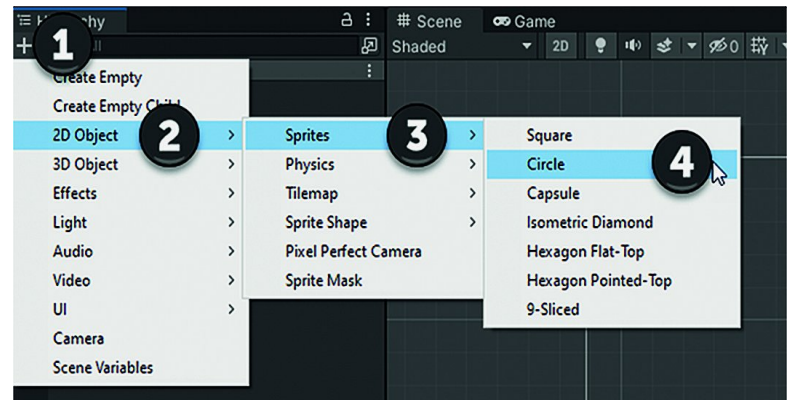
Nieuw object

Om een en ander nu tot leven te laten komen, want een

blauw vlak is wel wat saai, gaan we een nieuw object toevoegen. Dat kan in Unity op verschillende manieren:

1. Via de hierarchy
2. Via het menu
3. Via de assets

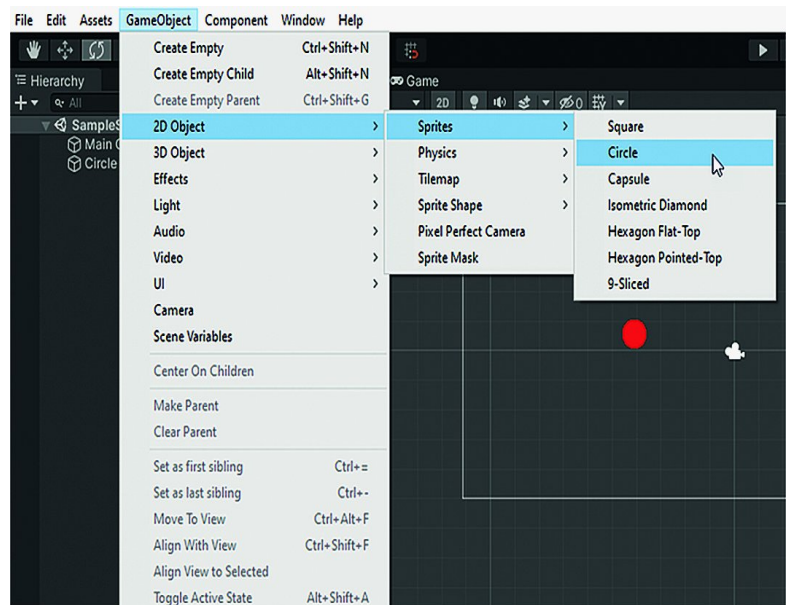
Via de hierarchy



Afb. 18 Via de hierarchy

Druk op de '+' bij (1), dus binnen de hierarchy. Dan willen we in dit geval een (2) tweedimensionaal (2D) object aanmaken en in dit geval een sprite (3) en wel een cirkel (4). Als je dit doet verschijnt een cirkel in de scene. Zelf selecteer ik de cirkel; ik kom er straks op terug, maar ik maak 'm even rood.

Via het menu



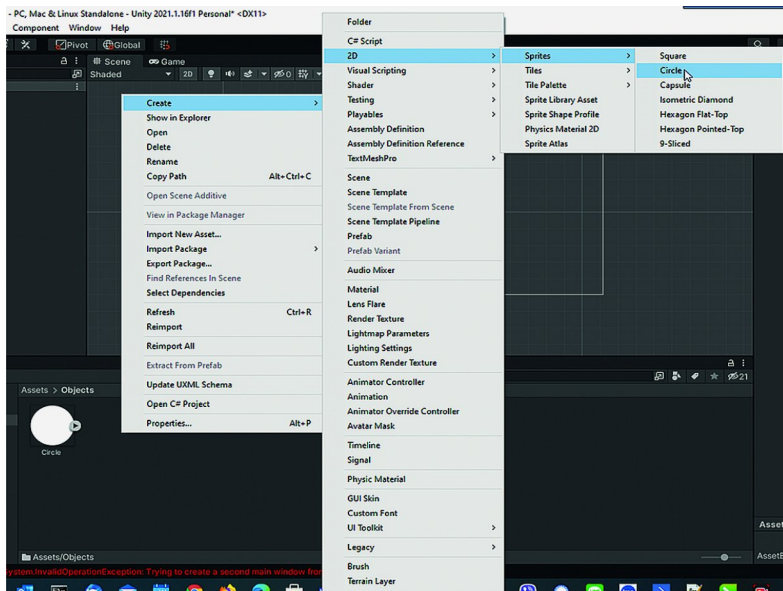
Afb. 19 Via het menu

Via het menu maak ik ook een cirkel aan. De gang van zaken is net als via de hierarchy. Voor de overzichtelijkheid maak ik deze cirkel groen.

Via de assets

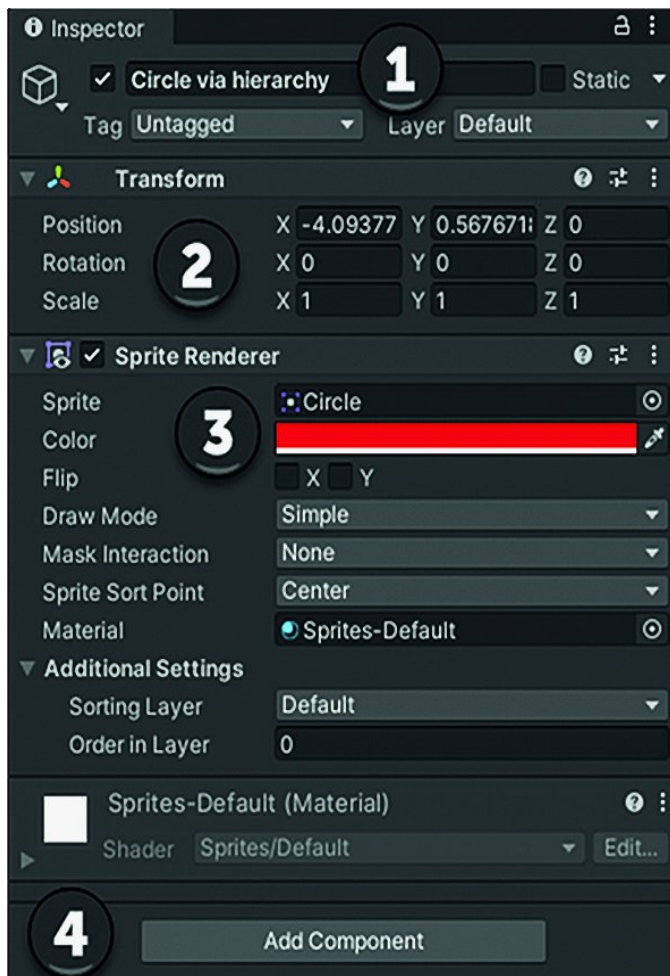
Binnen de assets hebben we nu een folder 'Scenes'. Als je ergens in de lege ruimte met je rechtermuisknop klikt, krijg je de gelegenheid om iets toe te voegen. Het is aan te bevelen, e.e.a. in aparte folders op te slaan, want het kan al gauw onoverzichtelijk worden als de games iets uitgebreider worden. Nu is het nog niet zo spannend, maar het is beter om gelijk goede gewoontes aan te leren.

Ik maak dus eerst een folder aan met 'Objects'. Je zou daarbinnen meer folders aan kunnen maken, zoals voor spelers, vijanden, enz., maar zover zijn we nog niet. Vervolgens selecteer je deze folder en binnen die folder klik je met je rechtermuisknop:



Afb. 20 Via de assets: asset aanmaken

Als het gelukt is zie je een cirkel in je assetbox. Deze slepen we naar de scene (of naar de hierarchy). Nu hebben we dus drie cirkels in onze scene, die uiteraard nog helemaal niets doen. Als we nu weer de camera selecteren, dan zien we die drie cirkels in het beeld. Laten we eens naar een van die cirkels kijken in de inspector:

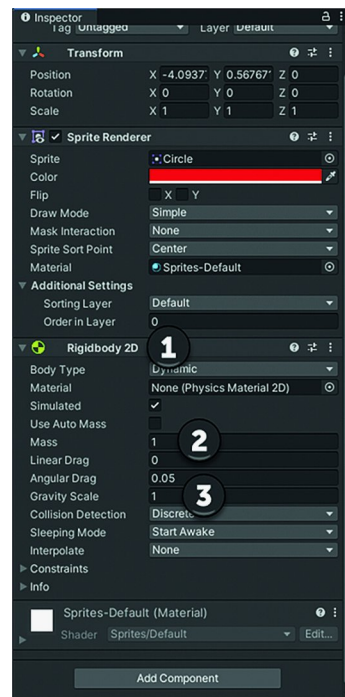


Afb. 21 De eigenschappen van de cirkel

Bij (1) zie je dat ik deze cirkel een eigen naam heb gegeven, om hem te onderscheiden van de andere cirkels. Bij (2) zie je de positie e.d. van deze cirkel. Ook al hebben we te maken met een tweedimensionaal object, toch hebben we ook een Z-waarde. Hierover een volgende keer meer, maar de driedimensionale wiskunde blijft van toepassing. In feite staat de Z-as nu loodrecht op het scherm, dus min of meer naar de kijker toe (of eigenlijk, ervan af).

Bij (3) kunnen we de kleur veranderen. Met het pipetje kun je een kleur van het scherm afhalen, zoals dat ook in andere programma's werkt. Door op de kleur te klikken kun je met de RGB-waarden of via een palet van alle kleuren de juiste kleur kiezen. De andere instellingen komen misschien in een later stadium aan de orde, maar waarschijnlijk niet eens allemaal. Er is zo veel in te stellen, maar niet alles is bedoeld voor beginners.

Bij (4) kun je een component aan dit object toevoegen en dat gaan we eens doen. Klik op 'Add component' en binnen het menu dat dan verschijnt klik je op 'Physics 2D' ofwel mechanica in een tweedimensionale wereld. We kiezen daarbinnen voor 'Rigidbody 2D' om de cirkel als een fysiek object in te stellen. Zodra je dat gedaan hebt, komt er een blokje binnen de inspector bij:



Bij (1) zie je dus de instellingen voor deze 'Rigidbody'. En hier kun je bijvoorbeeld (2) de massa en (3) de zwaartekracht instellen. Als we deze waarden ingevuld hebben, dan gedraagt deze cirkel zich als een bal in een tweedimensionale wereld en wil hij naar beneden vallen. En als deze bal nergens op valt, dan blijft hij maar doorvallen. In de scene zie je het zichtbare vlak afgebeeld, het gedeelte dat je met de camera kunt zien. Maar de wereld binnen Unity is vele malen groter, in principe uitsluitend beperkt door het geheugen van de computer, dus zolang de bal nergens op valt, blijft hij vallen. Probeer maar, druk op de play-knop bovenin het scherm. We hebben alleen de rode bal deze eigenschap

gegeven, dus alleen de rode bal valt.

Het artikel is door de vele screenshots wat groter uitgevallen dan je van één artikel mag verwachten. Daarom laat ik het nu hierbij en gaan we volgende keer verder met het laten stuiteren en opvangen van de ballen, om een gevoel voor de bewegingen binnen Unity te krijgen.

Maar besef wel, we hebben nu dus al een bal laten bewegen in een scherm, zonder een regel te programmeren. En volgende keer gaan we ook nog niet programmeren, maar zetten we wel veel meer in beweging en gaan we nog meer leuke dingen doen.

De volgende artikelen willen we beperken tot de gebruikelijke vier pagina's, zoals de laatste tijd met Scratch ook redelijk lukt, maar ik had deze keer wat veel screenshots nodig om tot een goede uitleg te komen.

Ik hoop dat ik jullie enthousiast heb gemaakt om hiermee verder te gaan en ik nodig de liefhebbers uit om contact te leggen via e-mail: gameontwerp@compusers.nl.

Zoals gezegd aan het begin, dit adres is intussen actief en wordt met enige regelmaat ook uitgelezen.