# Teknologianalyse

## Valg af spilmotor

Det er af afgørende betydning for virksomhedens fremtid, at vi vælger det rigtige udviklingsværktøj fra starten. At skifte fra en spilmotor til en anden er ensbetydende med et dyk i firmaets akkumulerede KnowHow, og en periode med indlæring af nye systemer, som ellers kunne være brugt på udvikling.

Der findes naturligvis en lang række andre, både proprietære og kommercielle spilmotorer, men for at bevare overskueligheden fokuseres der udelukkende på de følgende da de repræsenterer mest udbredte og veldokumenterede.

### CryEngine

|  |  |
| --- | --- |
| CryEngine 4 |  |
| Versioner: | Gratis til uddannelse og non-kommerciel brug. Ellers individuelt aftalt licens. Abonnementsordning på vej: 50kr/md |
| Unikke Features: | Bliver ofte fremhævet for dens håndtering af tung grafik. |
| Scripting sprog: | LUA |
| Understøttede platforme: | PC, Xbox 360, Xbox One, PlayStation®3, PlayStation®4 or Wii U. |
| Fokus: | Tung grafik, store teams, store projekter, PC og konsol |

CryEngine udvikles af tyske CryTek, og har i udgangspunktet kun været brugt af store og veletablerede firmaer. Grunden dertil skal formentlig findes i, at et spilfirma skal have en vis størrelse for at kunne retfærdigøre udgiften til en licens. Derudover har CryEngine bygget sit ry på at levere de, subjektivt, flotteste spil i branchen. Det har værdi for AAA-udviklere, men er nok mindre væsentligt for et lille spilhus.

Det er ikke til at sige hvad en pro-licens helt præcist koster, da priserne varierer efter størrelsen på det firma der henvender sig. Der er dog tegn der tyder i retningen af, at CryTek skal have 20% af bruttofortjenesten ved licenser givet til indie-udviklere.  
Dog har CryTek for nyligt annonceret, at de vil implementere en abonnementsplan til ca. 50kr. om måneden, og uden at kræve royalties for udgivne titler. Det er endnu ikke officielt annonceret om der bliver adgang til kildekoden, eller om der er andre restriktioner der skal tages højde for. Umiddelbart gør det dog at det med et slag er blevet den billigste af de undersøgte spilmotorer.  
Der er et aktivt fællesskab omkring CryEngine, og der er adgang til forskelligt undervisningsmateriale. Men af de spilmotorer som der her tages udgangspunkt i, er det den der virker til at have det mindste fællesskab. Konkret betyder det, at udviklingen af spil vil tage længere tid, og medføre flere frustrationer, end det ellers havde været tilfældet.

### Unity

|  |  |
| --- | --- |
| Unity3D |  |
| Versioner: | Gratis udgave (inkl. Gratis iOS og Android moduler)  Pro-version til 8000kr. (abb. 400kr/md)  Pro-iOS: +8000kr. (abb. +400kr/md)  Pro-Android: +8000kr. (abb. +300/md) |
| Unikke Features: | Massiv brugerbase giver svar på alt |
| Scripting sprog: | C#, UnityScript (JS mod), Boo |
| Understøttede platforme: | PC, Mac, Linux, Xbox 360, Xbox One, PlayStation®3, PlayStation®4 or Wii U, WebGL, iOS, Android, Windows Phone, BlackBerry. |
| Fokus: | Generel spiludvikling, mindre teams, og mindre projekter. All-round devices |

Indtil tidligere I år var der reelt kun én spiller på markedet for virksomheder af vores slags, og det var Unity. Derfor er der også et massivt community omkring Unity-motoren, og da der er rigtigt mange nybegyndere betyder det at mængden besvarede spørgsmål omkring brugen af spilmotoren, og spiludvikling i det hele taget er massiv. Dertil kommer, at Unity selv er meget aktive med udviklingen af læringsmateriale. Det er også et absolut plus, at det er relativt let at portere et spil fra en platform til en anden, og Unity har pt. klassebiblioteker til alle de platforme som er relevante for virksomheden.  
Med en pris på nul kroner og nul øre for gratis-versionen, der indeholder langt de fleste af systemets komponenter, er der intet at sige til at mange nybegyndere har taget Unity til sig. Paradoksalt nok betyder det til gengæld, at det kan være svært at finde decideret læringsmateriale om mere avancerede emner.

Der er ikke mulighed for at redigere direkte i Unity´s kildekode, men man kan tilføje sine egne komponenter, og af den vej bøje systemet til sin vilje.

Prisen på pro-versionen er der ikke så meget at sige til, det koster 8100 for en licens til den cyklus Unity er i, og den ejer man så til evig tid. Hvis man vil opgradere til den efterfølgende cyklus koster det det halve. Nu, og indtil Unity 5 lanceres kan man pre-order version 5, og få adgang til Unity 4 pro indtil da.  
Hvis man vil udvikle til smartdevices koster det yderligere 8100 for iOS pro og igen 8100 for Android Pro.   
Man får altså ikke adgang til f.eks. iOS pro features selvom man har Unity pro.   
Det er der mange meninger om, men det betyder at det hurtigt kan eskalere i prisen. Især fordi det der er tale om her gælder én licens. Hvis et team er på 4-5 mand, hvilket er typisk, skal man altså gange op.  
Unity har også en abonnementsordning hvor man kan leje hver enkelt del for 400 kr./md.   
Benytter man sig af denne løsning så er det værd at bemærke, at man binder man sig for et år af gangen.

### GameMaker

|  |  |
| --- | --- |
| GameMaker Studio |  |
| Versioner: | Flere versioner, med forskellig pris efter features  Fra gratis til 4300kr. |
| Unikke Features: | Lav et spil helt uden programmering |
| Scripting sprog: | GML (Game Maker Language) |
| Understøttede platforme: | PC, Mac, Android, HTML5, iOS, Linux, Windows Phone, Tizen |
| Fokus: | For den absolutte nybegynder, Små projekter, PC, Web, mobil. |

GameMaker udvikles af YoYo Games, er mest at betragte som nybegynderens første værktøj. Som bruger kan man udvikle et spil uden at skulle skrive en eneste linie kode. Af den grund er det vidt udbredt blandt hobbyister. Alligevel er der eksempler på succesfulde spil der er udviklet med GameMaker, senest Hotline Miami, Spelunky og Risk of Rain.   
Omvendt er der ingen større titler på deres showcase liste, hvilket indikerer at der er nogle begrænsninger i den pipeline som systemet er bygget op over.

Selvom det er muligt helt at undgå kode, så er der mulighed for selv at scripte ved brug af det javascript-lignende GML-sprog.  
  
Selvom GameMaker kan fås fra 0kr, så kræver det Master udgaven til 4300 kr, for at kunne eksportere til iOS og Android.

I lighed med Unity er der et stort fællesskab omkring systemet, hvilket væsentligt bidrager til at stabilisere læringskurven.

Unreal Engine

|  |  |
| --- | --- |
| Unreal Engine 4 |  |
| Versioner: | Individuel prissætning. Abonnementsordning  100kr/md + 5% af brutto |
| Unikke Features: | Fuld adgang til kildekoden. |
| Scripting sprog: | C++ |
| Understøttede platforme: | PC, Mac, iOS, Android.  Konsoller er ikke omfattet af abonnementsordningen. |
| Fokus: | Store teams, store projekter, PC og konsol |

Unreal Engine udvikles af Epic Games, og har navn efter det klassiske FPS Unreal fra 1998. UE har været anvendt i en meget lang række spil siden da, bla. Mass Effect og Gears of War-spillene. Selv om den oprindeligt primært var til brug for FPS-spil, viser listen at det har været muligt at udvide den og lave tilpasninger så den har kunnet bruges til spil i alle genrer.

Indtil nu har spilstudier kunnet forhandle sig til en licens hos Epic. Der er ikke angivet faste priser, så man kan kun spekulere i hvad det har kostet.  
Siden 2009 har man kunnet hente en gratis ”light”-udgave af Unreal Engine 3, kaldet UDK (Unreal Developer Kit). Hvis man udviklede og udgav noget ved brug af den, har det kostet et engangsgebyr på 535kr. samt 25% af bruttofortjenesten.

Ligesom tilfældet var med CryEngine, er det tidligere på året blevet annonceret, at Unreal Engine 4 bliver tilknyttet en abonnementsordning, hvor man for godt 100kr/md kan få en fuldt opdateret version af spilmotoren. Dog skal man også aflevere 5% af bruttofortjenesten på ens udgivelser.   
Hvis man benytter sig af denne model kan man udgive til Mac, PC, iOS og Android, men ikke konsoller. I givet fald skal man igen tage kontakt til Epic for en handel.

Epic stiller dokumentation og læringsvideoer til rådighed, og der er også et dedikeret fællesskab hvor man kan finde løsninger. Og, i modsætning til de andre systemer er der her mulighed for at redigere direkte i motorens kildekode.

### Byg eget framework

Den sidste mulighed er naturligvis, at firmaet går i gang med at udvikle sit eget framework. Det ville medføre 100% fleksibilitet i forhold til firmaets fremtidige udvikling. Samtidigt kan en god spilmotor generere massiv indtægt til virksomheden, hvis den kan sælges eller lejes ud til andre udviklere.  
Men det er en massiv opgave at give sig i kast med. Det kræver et højt teknisk niveau, og det kræver kapital der kan understøtte virksomheden indtil den er klar til brug. Det kan tage flere år at nå til det punkt.

### Konklusion

Selvom både CryEngine og Unreal Engine 4 er billigere i anskaffelse end Unity (For den fulde version), så er konklusionen at firmaet bør anvende netop Unity. Det skyldes at fælleskabet omkring dette system er enormt, og beviseligt gerne deler ud af dets erfaringer. Der er ikke det spørgsmål som ikke er blevet stillet, og besvaret, og det alene er nærmest nok til at vælge Unity frem for nogen andre.

Dertil kommer, at alle de involverede i virksomheden allerede har stiftet et indgående bekendtskab med Unity. Det er ikke nogen triviel opgave at blive omskolet til en ny spilmotor, så det trækker også væsentligt i retning af unity.

De sidste par måneders røre, som er forårsaget af at både Epic og CryTek har annonceret at deres systemer bliver tilgængeligt i et prisleje hvor selv de mindste kan være med, betyder dog at det ikke er sikkert at virksomheden skal binde sig alt for meget til en platform.  
Netop den type overvejelser er en vægtig grund for at udvikle sit eget system, om det så er et framework, eller en decideret spilmotor. Hvis man har muligheden for det, så har man også selv al kontrollen.  
Det er dog ikke en realistisk mulighed for dette firma, da ingen af de involverede har erfaring med den type udvikling. Selv i bedste fald ville der gå årevis før en egenudviklet spilmotor ville kunne bruges til seriøs udvikling, og i den tid ville der ikke være nogen indtægter ej heller nogen mulighed for virksomheden for at positionere sig på markedet.

Angående GameMaker, så er det måske nok et system som vi bør afprøve, da det muligvis kan være tidssparende i forhold til udvikling af prototyper. Som primær spilmotor virker det dog afskrækkende, at der ikke findes eksempler på store produktioner i deres Showroom. Det er ikke i virksomhedens bedste interesse at binde sig op på en motor hvorfra vi ikke kan skalere til større produktioner.

Altså bør virksomheden udvikle det første spil med Unity, og i øvrigt forsøge at holde sig til gratisversionen frem til udgivelsen, og så i den afsluttende fase benytte den 30-dages gratis pro-version.

## Valg af billedredigeringsprogram

For at kunne bygge spil, skal vi bruge en billededitor, jf. vores afgrænsning er det ikke et område som vi vil dykke dybt ned i. Vores grafiker er vant til at arbejde med Blender til 3D-modeller og animationer, og Gimp til billedredigering, og da det samtidigt er gratis-produkter, så det er oplagt at fortsætte med at benytte disse værktøjer.  
  
Ikke desto mindre er vi opmærksomme på, at vi der er alternativer som vi kan få brug for på længere sigt.   
Den åbenlyse kandidat i forhold til billedredigering er Photoshop, som er en standard i industrien. Ud over Photoshop producerer Adobe også en række andre interessante produkter, som f.eks. Illustrator, som vi kunne have gavn af.  
  
Hvad angår 3D-modellering, så er det den samme situation, Blender er godt nok til vores umiddelbare behov, men mere professionelle produkter som f.eks. Maya og 3D Studio Max har flere og bedre funktioner end Blender, og derfor kan det på den længere bane være interessant at lave en mere konkret analyse.   
For nuværende er det dog ikke interessant, fordi behovet for den type produkter ikke eksisterer.

## Versionsstyring

For at sikre, at vi har fuld kontrol over vores produktion, og ikke er i farezonen for at miste data, er det afgørende at vi har versionskontrol på vores projekt. Der findes en lang række mulighed

### Unity Team Server

For det første har Unity selv et ”Team-License”-produkt der kan integreres direkte i Unity. Det koster 2700 kr. pr person, og der er ikke nogen Sky-funktion inkluderet, hvilket vil sige at vi selv skal have en server sat op.  
Til gengæld er Unity produktet skræddersyet til at håndtere både den type af filer som Unity producerer, og den størrelse de kan forvente at komme i.

### GitHub Alternativt kan vi betale os fra en privat konto hos f.eks. GitHub. De kan fås fra 40 kr. om måneden. Til gengæld begrænser vi os til projekter der er mindre en 1GB, og filer som ikke overstiger 100MB. Ligeledes anbefaler GitHub at man bruger et andet redskab til ”mellemregninger” af tungere typer filer, som f.eks. 3D-modeller. Strengt taget behøvede vi slet ikke betale for en GitHub konto, men gratis-kontierne er offentligt tilgængelige. Det er ikke noget problem i første omgang, men det er usmart hvis vi skal arbejde med en tredjepart, eller generelt vil holde et projekt lidt tættere til kroppen.

### Bitbucket

Bitbucket er et alternativ til GitHub. I modsætning til GitHub er det muligt at få private repositories gratis for op til 5 brugere. Samtidigt kan vi have et ubegrænset antal repositories, hvilket vil sige at så længe firmaet har max 5 personer tilknyttet der skal bruge adgang, så kan vi få gratis opbevaring. I øvrigt er der integration med Jira indbygget, hvilket er relevant at tage i betragtning når vi kommer til et tidspunkt hvor vi får brug for et reelt bugtracker-værktøj.  
I forhold til projekt- og filstørrelser, så er det i store træk de samme regler der gør sig gældende her, som hos GitHub.

### Dropbox

Vi kan også benytte os af en ressource som Dropbox. Dels er det let at dele materiale imellem projektdeltagere, men det er også muligt at sætte et versionsstyringssystem op direkte i en mappe. Igen kan man benytte sig af gratisløsningen hvor hver bruger har 2GB (I udgangspunktet), men det er ikke stort bedre end med GitHub.  
Den optimale dropbox løsning ville koste 400 kr./md.

### Konklusion

Der er flere alternativer, som sætte et system op. Men i forhold til opstarten af virksomheden så er det at skyde langt over målet. Hvis vi selv satte en server op, så ville det også medføre at vi selv skulle stå for vedligehold af den, håndtere back-ups og opgraderinger og meget mere. Vi er et spilfirma, ikke en hostingservice. Eller sagt på en anden måde, det er ikke et område vi vil bruge vores tid og energi på, det skal bare virke.

Med alt dette i mente har vi besluttet at bruge en kombination af Dropbox og GitHub til vores første projekt. Alle midlertidige assets gemmer vi i vores, gratis, Dropbox mapper, og de bliver først importeret til vort Unity projekt når de er klar til det. Selve Unity-projektet er decideret versionsstyret og ligger på en gratis-konto på GitHub. Det er ikke specielt relevant lige nu, at det er offentligt tilgængeligt, for det er svært at forestille sig, at nogen skulle være interesserede i det. Hvis det bliver vurderet som nødvendigt, kan vi slette det Git Repository som projektet ligger i, når spillet er klar til udgivelse.

Af selvsamme grund kunne Bitbucket være et relevant produkt til det næste projekt. Når vi har valgt det fra i denne omgang, så skyldes det at vi har erfaring med at bruge GitHub, og det er ikke et område vi er interesserede i at bruge energi på i denne ombæring.  
Reglerne angående fil- og projektstørrelse behøver vi i første omgang ikke bekymre os om, da det er er urimeligt at forestille sig vores projekter overstige 1GB. Når vi kommer til lidt større projekter, så er det et emne der kan blive taget op igen.  
Som klient til håndtering af versionsstyringen har vi valgt at bruge SourceTree, ud fra den betragtning at det er gratis at bruge, og vi har erfaring med det fra tidligere projekter.