

## Seminariedokumentet

### Lärandemål:

Jag vill lära mig mer om procentuell generering ur ett level design perspektiv. Jag vill titta på två sätt att generera en slumpmässigt skapad level design. Där är ett fokuserar mer på programmering och att generera banan genom kod, medan det andra skulle fokusera mer på att skapa designade delar och hur de sätts samman på ett sömlöst sätt. Jag vill gärna jobba med level design i framtiden, och ser detta som en bra övning i hur man kan skapa slumpmässiga banor.

### Aktiviteter/genomförandemål:

Jag har en redan en prototyp där jag har problemet med hur jag ska generera banan. I denna prototyp kommer jag att implementera de två olika sätt en. Det minsta jag vill göra är att implementera ett av de två sätten. Utöver praktiskt arbete kommer jag även att behöva forska kring procentuell generering, tex vad det finns för olika typer. Jag kommer att jobba i Unity3D vilket är det verktyget jag är mest bekväm med. Det är nog inte nödvändigt att skapa 3D assets för spelet, då det går lika bra att använda de assets som redan finns i Unity. Men om jag skulle fastna i programmeringen kan det vara bra att byta aktivitet. Om så är fallet kanske jag skapar grundläggande assets för prototypen.

### Tidsplanering:

Jag har ingen exakt tidsplan över hur jag kommer arbeta. Min generella plan är att spendera en vecka på att undersöka procentuell generering. Jaa kommer förmodligen också behöva tid att sätta upp mitt Unity projekt, det är några mekaniker jag skulle vilja färdigställa, som jag inte är super nöjd med i nuläget. Sedan spendera halva av resterande tid på att först göra den modulärt procentuellt genererade banan och den andra halvan på procentuell generering genom kod. Om jag inte blir klar med den modulära procentuella genererade banan, kommer jag fortsätta att jobba på den tills den är klar och efter det börja på procentuell generering genom kod.

### Riskanalys:

Den största risken är att jag inte kan genomföra att skapa banorna pga bristande programmeringskunskaper. Det är en ganska stor sannolikhet att det kommer bli svårt, men om man tar saker i små steg och frågar om hjälp så går det säkert bra.

### Hållbar utveckling:

Att använda procentuell generering kan vara ett bra sätt att minska arbetsbördan för en designer med underhåll och att behöva skapa nya banor som måste designas och speltestas. Om de skapas slumpmässigt är det mindre underhåll för en designer.

## Feedback till seminariet

Feedback från Seminariet:

Jag fick feedback om att utveckla riskerna och kopplingen till hållbar utveckling, samt att projektet kändes för stort.

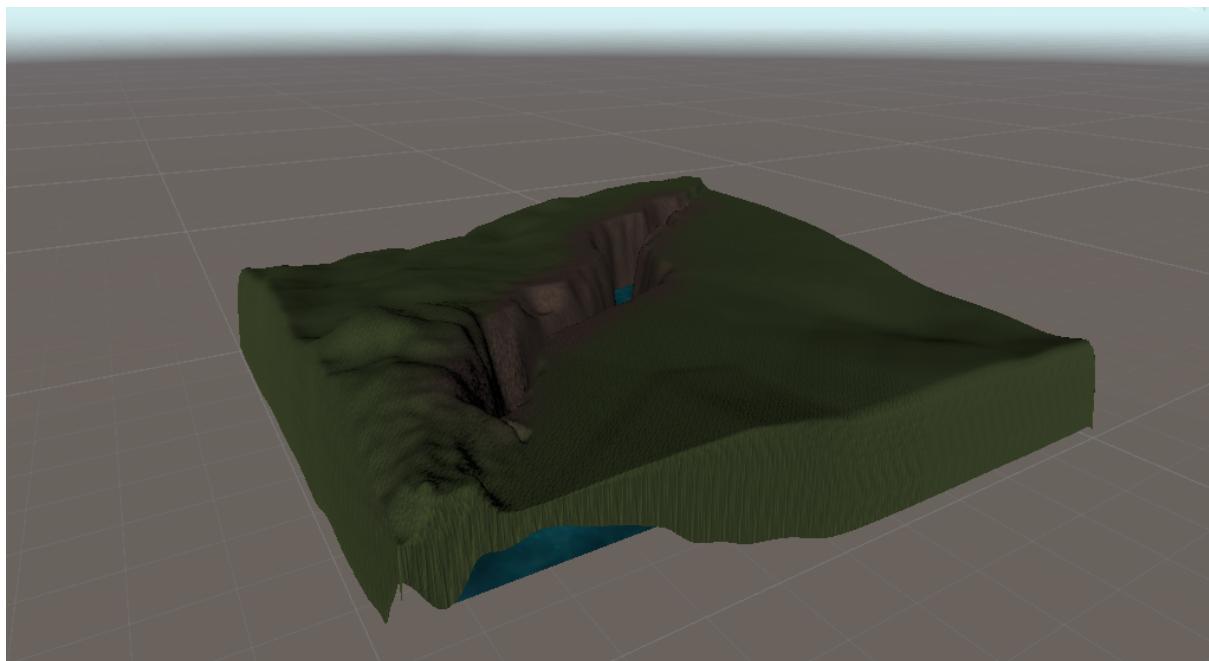
Jag ser inte så stora risker med projektet då det är en lärande process. Klart att det kan hända att material och arbete försvisser, även att brist på kunskap kan göra att jag inte kan uppnå målen jag sätter upp. Dock anser jag inte att det kommer förhindra mitt lärande hur jag ska uppnå de målen. Genom att misslyckas eller behöva göra om kan man också lära sig. För de riskerna jag tog upp i originaldokumentet kan jag fråga på forum online om hjälp.

För kopplingen till hållbar utveckling, det förekommer en hel del crunch i spelutvecklingen. Om man kan minska arbetsbördan genom att använda verktyg kan man minska utslitning av arbetarna. Detta är ju inte det enda man behöver göra för att lösa crunch. Då redan i industrin finns en förväntar sig att man ska jobba över 40 timmar i veckan. Så attityden hos många behöver även ändras men man måste börja någonstans för att uppnå ett hållbart sätt att utveckla spel.

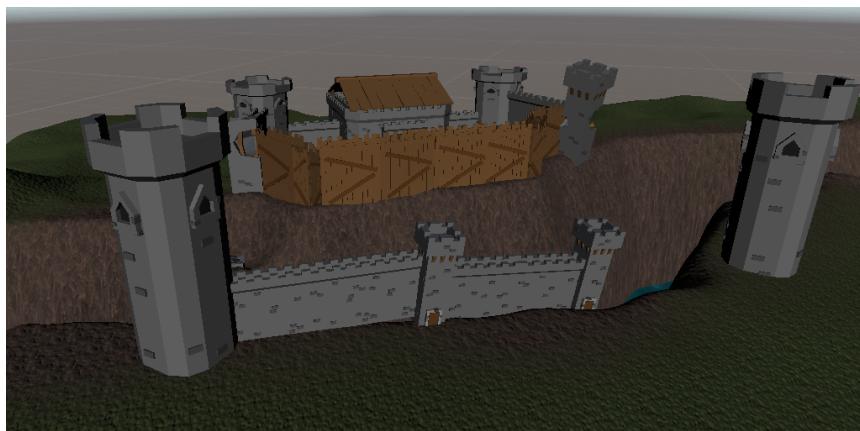
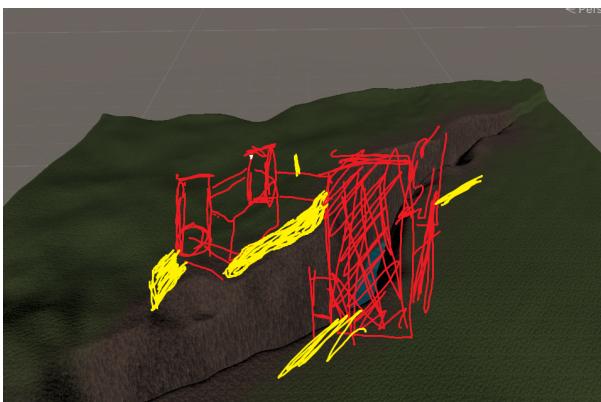
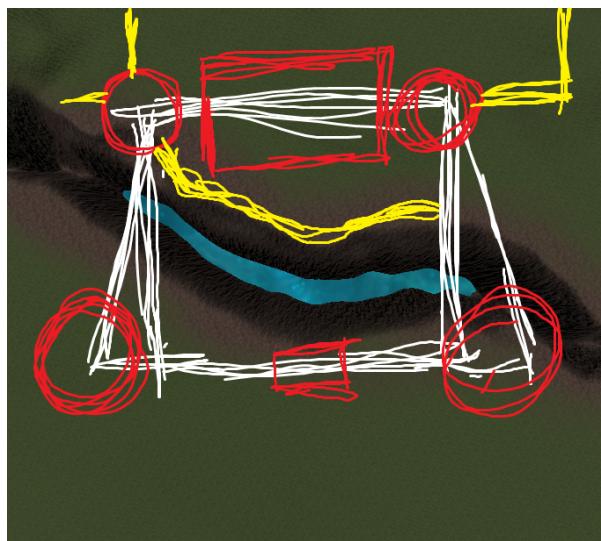
## Seminarium - Avstämning

Material:

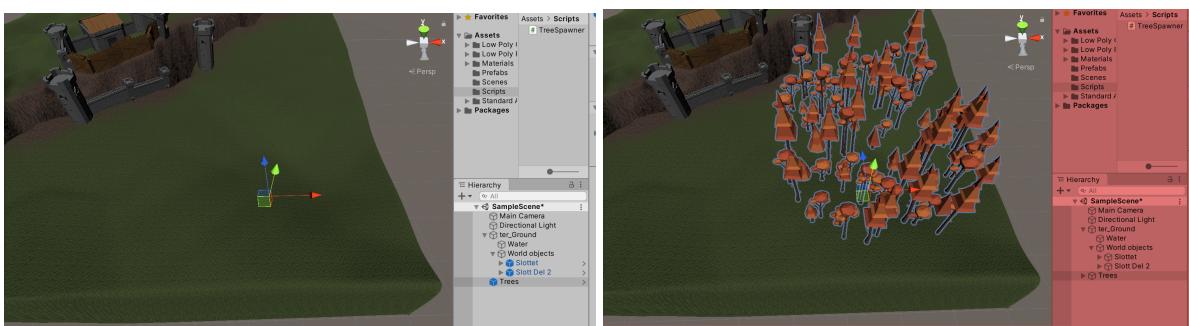
Skapat en terräng som en grund:



Icke procedurella generade object:



### Procedurell Generering av träd:



```

private Vector3 TreePosition()
{
    var circleSize = Random.insideUnitCircle * radius;
    Vector3 worldPos = transform.TransformPoint(circleSize.x, 0, circleSize.y);

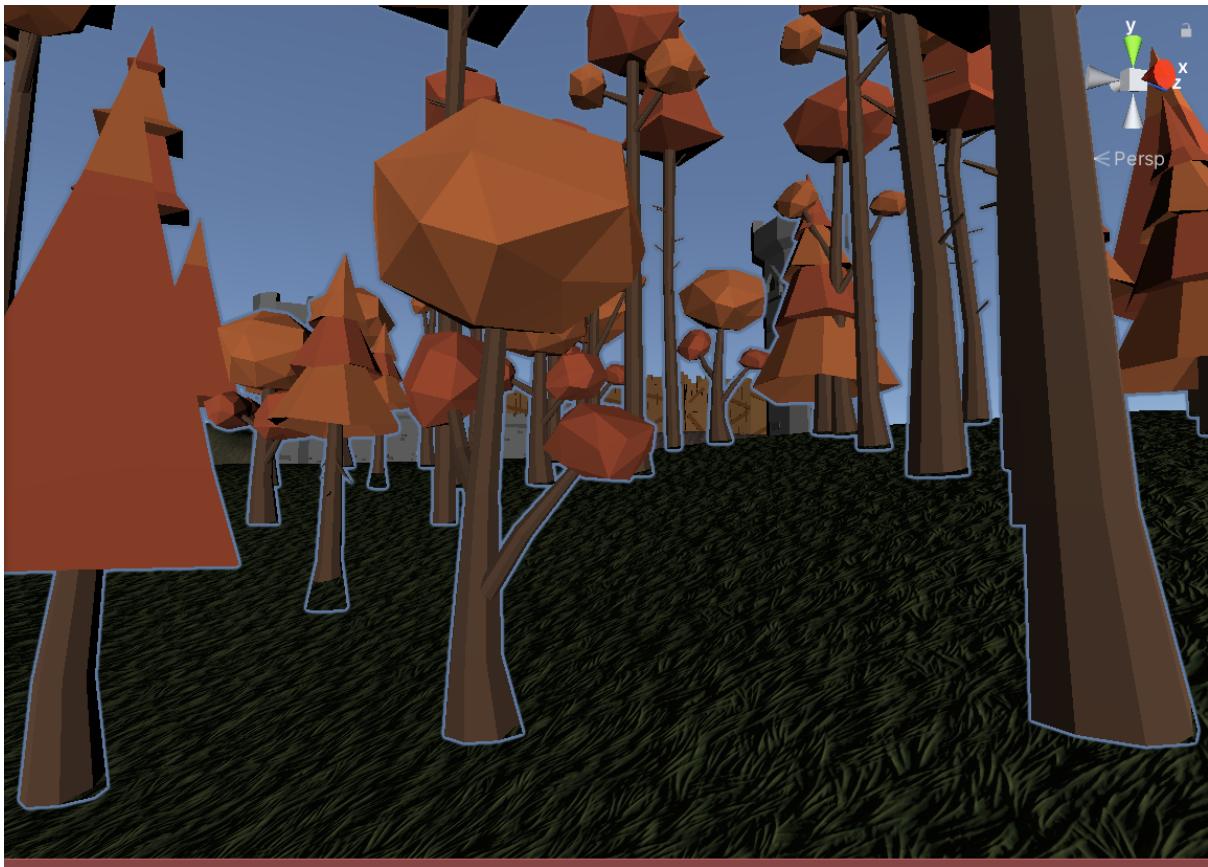
    RaycastHit hit;
    float terrainHeight = 0f;
    int layerMask = 1 << 8;

    //kolla upp hur raycast fungerar.

    if (Physics.Raycast(new Vector3(worldPos.x, 9999f, worldPos.z), Vector3.down, out hit, Mathf.Infinity, layerMask)) {
        terrainHeight = hit.point.y;
    }

    float _pointSpawn = terrainHeight;
    var treePosition = new Vector3(circleSize.x, _pointSpawn, circleSize.y) + transform.position;
    return treePosition;
}

```



### Källor:

Jag har för att lära mig om procedurell generering titta på hur många andra spel använder procedurell generering tex Spelunky och Minecraft. Utöver det läste jag två artiklar, en kring hur man använder procedurell generering på effektiva sätt och den andra går igenom hur man skapar procedurell generering i Unity genom Perlin Noise.

*How (and Why) Spelunky Makes its Own Levels | Game Maker's Toolkit*

*Brian Walker - Procedural level design in Brogue and beyond.*

*(HOW do MINECRAFT WORLDS GENERATE? & How Minecraft Generates Such HUGE Worlds | Game>Show | PBS Digital Studios)*

*How to effectively use procedural generation in games*

*Complete Guide to Procedural Level Generation in Unity – Part 1*

Källorna jag kollat på

Dessa källor har gett mig en bra överblick hur jag kan använda procedurell generering. nämligen att jag kan bestämma var producera genereringen ska ske, så att det inte ser överallt. När det kom till att skapa koden för den producera genereringen undersökte jag hur andra hanterar att implementera träd på terräng, där kom jag över en video av Holistic3d som har en serie av olika typer av procedurell generering i specifikt Unity. Jag lärde mig mycket från hennes videos och kunde utifrån det skapa mitt egen kod för procedurell generering. Jag fastnade enbart när det kom till att placera objekt på terrängen, men använde UnityChat's video Unity RTS World Builder [2] - Random Position Instantiation vilket löste det.

Efter som jag har fokuserat mycket på hur den procedurella generering ska funka har jag inte läst så mycket som leveldesign och olika teorier inom just det. Men tänker att jag kommer göra det när jag är nöjd med den procedurella genereringen.

Procedural Generation in Unity

[https://www.youtube.com/playlist?list=PLi-ukGVOag\\_0vJMJKAjUyuPF3kMXKW2IV](https://www.youtube.com/playlist?list=PLi-ukGVOag_0vJMJKAjUyuPF3kMXKW2IV)

Unity RTS World Builder [2] - Random Position Instantiation

<https://www.youtube.com/watch?v=MVeI9y6k8MU>

Videos om procedurell generering

Status:

Jag har bytt artefakt som jag skaper från ett spel med procedurell generering, till en miljö där bara skogen är procedurellt genererade. Bytet kom från att jag kände att min första ide hade för mycket fokus på kod och att få det att fungera, samt saknade en stark koppling till leveldesign.

Detta har lett till att planeringen har ändrats. Jag jobbar bara på en artefakt nu. Mitt minimimål är att procedurellt genererade en typ av träd på terrängen och utan att något object svävar i luften. Jag har under veckan lyckats med att uppnå detta mål. Det jag planerar att göra under fortsättningen av mitt arbete är att fortsätta utveckla hur den procedurella genereringen fungera tex. sannolikhet av att object skapas eller kanske att object inte spanar på varandra. Jag kommer också börja fokusera mer på leveldesign och söka upp några artiklar kring teorier som kan hjälpa mig skapa en miljö.

Riskanalys:

Jag tycker att min riskanalys fortfarande stämmer, jag har än så länge lyckats lösa mina problem genom att läsa på forum. Att jag har stött på problem har även hjälpt mitt lärande. När jag stötte på problem med att y värdet på träden i spelet inte matchade höjden på terrängen, fick jag söka på hur andra hade löst problemet. Då de använde Raycast fick jag lära mig hur de fungerade för att kunna skapa min kod.

Jag tror dock en risk jag glömt att tänka på är att min dator inte är jättebra och klara själva genereringen om det skulle bli för mycket. Min dators grafikkort är också skräp så jag kanske inte kan uppnå hur jag vill att den slutgiltiga miljön ska se ut pga det.

Jag kan inte göra så mycket åt grafikkortet, men jag vet att Unity har funktioner för att hjälpa med optimering och att den bara skapar objekt kameran kan se som jag skulle kunna fördjupa mig i.

#### Hållbar utveckling:

Jag fick feedback på förra seminariet att förtydliga min koppling till hållbar utveckling. Det förekommer en hel del crunch i spelutvecklingen. Jag tror att bland annat procedurell generering kan hjälpa förebygga crunch, jobba smartare inte hårdare. Om man kan minska arbetsbördan genom att använda verktyg kan man minska utslitning av arbetarna. Detta är ju inte det enda man behöver göra för att lösa crunch. Då redan i industrin finns en förväntar sig att man ska jobba över 40 timmar i veckan. Så attityden hos många behöver även ändras men man måste börja någonstans för att uppnå ett hållbart sätt att utveckla spel.

Fråga: Borde jag göra en karaktär (ish) så man kan gå runt i världen och kolla från ett första person perspektiv? För- och nackdelar?

#### Feedback:

Kunna kontrollera storlek i inspektorn

Höjdskillnader och instansiering

Generera i Editor

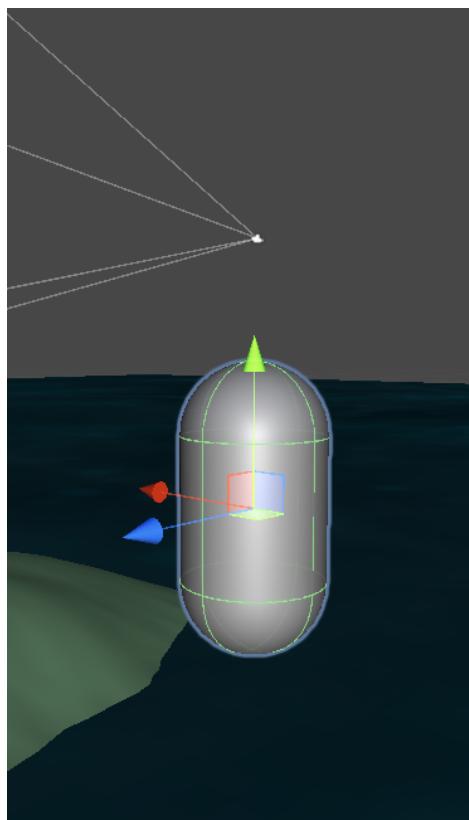
Skapa en karaktär

Dessa var några av förslagen jag fick från seminariet. De jag kommer fokusera på under fortsatt arbete är att beräkna höjdskillnader, generera i Inspektorn och göra en karaktär. Jag tror dessa skulle vara bra att lägga till i artefakten. Man kan redan kontrollera i storleken i skriptet och jag tycker det räcker.

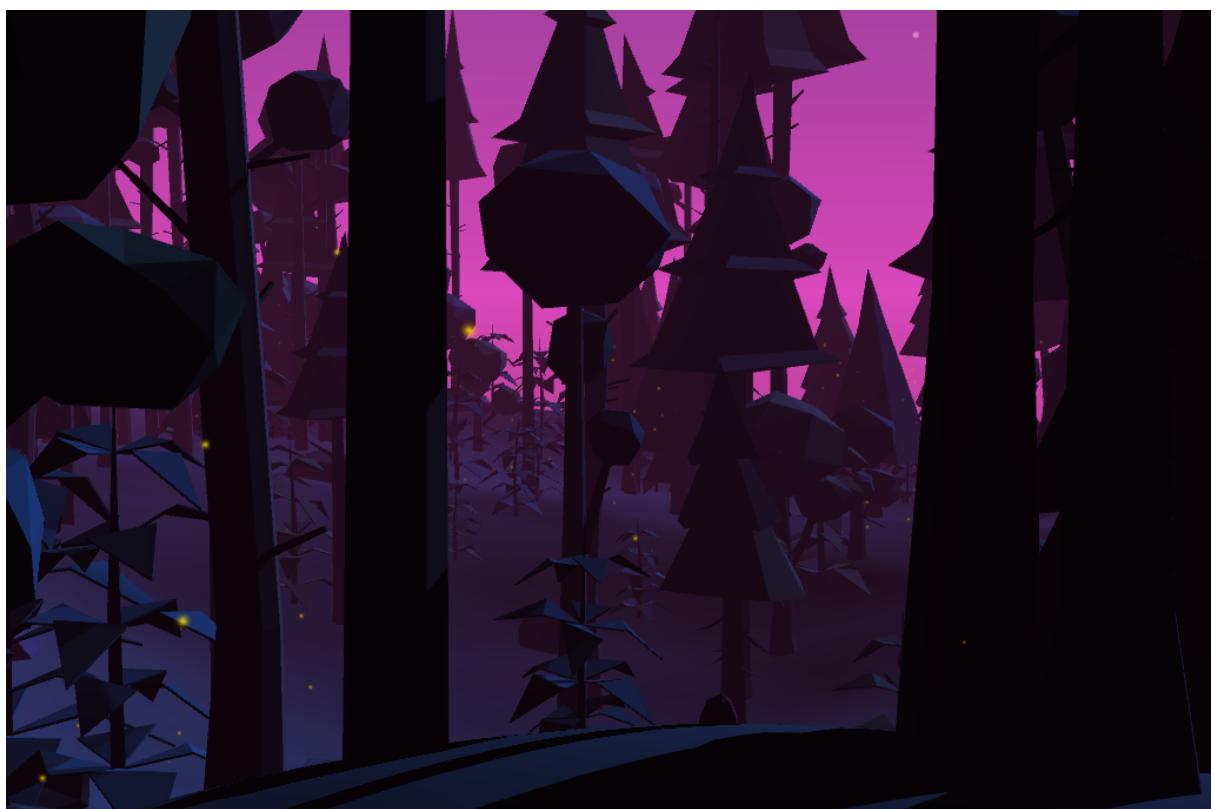
Under arbetet har jag lyckats implementera en karaktär och påbörjat att räkna ut höjdskillnader. Höjdskillnaderna är lite svårt men jag jobbar på det.

## Seminarium - Avrundning

Material:

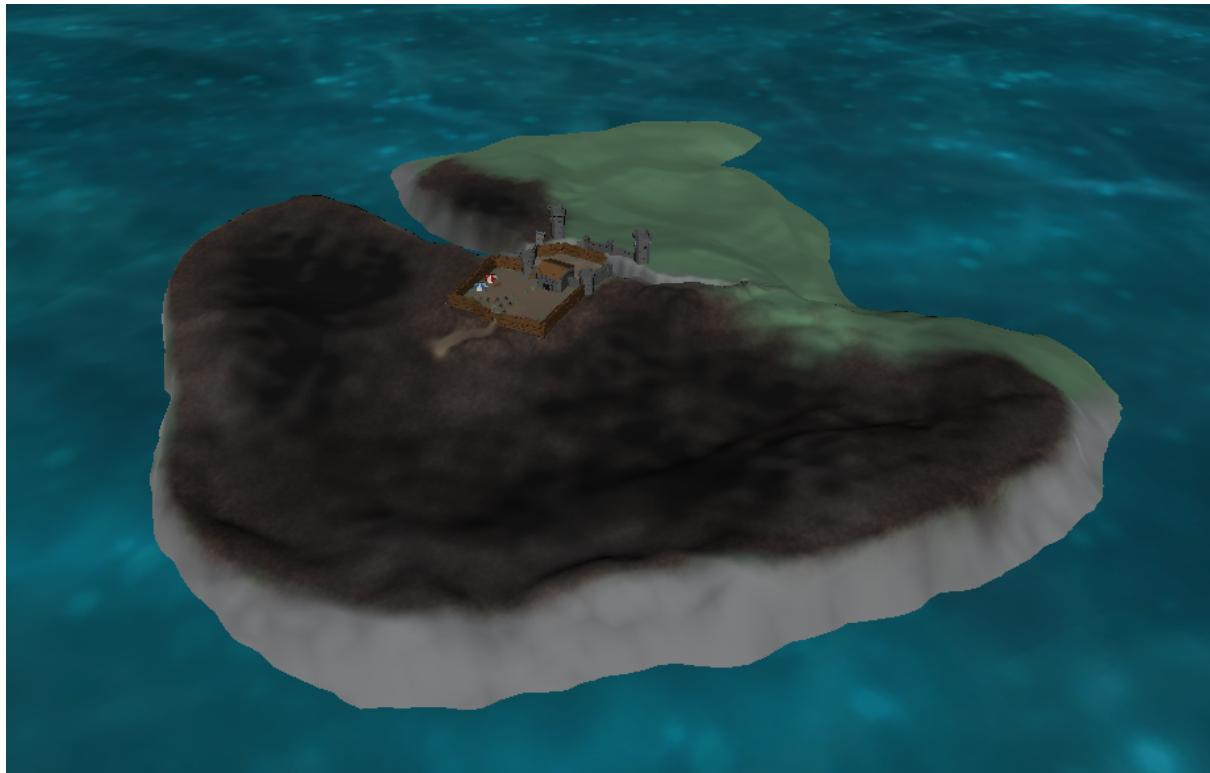
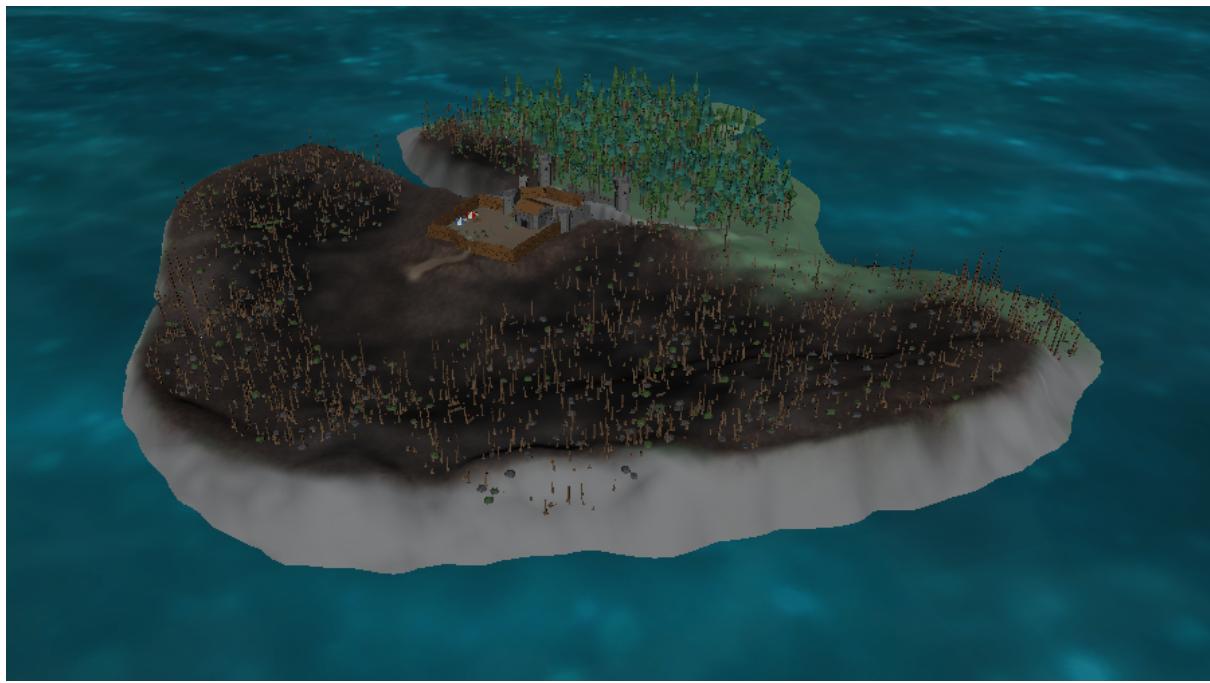


Spelar Object









### Resultat:

Jag har uppnått de minimemålen jag satte upp för min artefakt, samt att jag har lärt mig mer om procedurell generering. Jag tycker att jag har kommit ganska långt i mitt arbete utifrån målen. Jag är lite besviken dock på mig själv att jag inte har kunnat utveckla artefakten mer utifrån feedbacken från förra seminariet. Jag hade gärna implementerat stigar och att man kan mätta slutningar vid utplaceringen av objekt. Jag har tittat på att få in båda och har teorier om hur jag skulle kunna göra men mina programmeringskunskaper är lite för simpla för att lösa problemen.

tex. Har jag försökt att mätta slutningar, det jag försökt göra är att lägga in flera raycast som mäter området kring positionen där trädet skapas. Dessa värden läggs in i en Lista och jämför sedan värdena för de positionerna med ett värde som är höjdgränsen. Jag har inte fått och funka, men detta är min teori kring det.

### Kunskaper:

Jag har helt klart lärt mig mer om procedurell generering, men också ökat min kunskap kring unity och programmering. Så jag är nöjd med vad jag har lärt mig. Rörande procedurell generering, har jag lärt mig att det är olika svårt beroende på vad man vill göra. Att bara skapa en slumpmässigt genererad skog är relativt enkelt, medan att tex. generera en nivå baserat på var annan objekt hamnar är svårare och kräver mer logik. Även om jag kanske inte själv kan skapa komplexa genereringen så känner jag att jag har samlat tillräckligt mycket information om hur det kan fungerar, att jag kan kommunicera med programmerare och diskutera hur ett system skulle skapas både för att fungera effektivt och med bra leveldesign.

### Lärandeprocess:

Jag har lite blandade åsikter kring om min artefakt gjorde att jag uppnådde målen, jag lärde mig mer från att söka och läsa artiklar än vad jag gjorde när jag väl implementerade procedurell generering i mitt Unity projekt. Jag lärde mig mer om koden jag skrev och olika funktioner i Unity än procedurell generering. Men det var ett av mina sekundära mål så jag är inte så upprörd över det, jag upplever det som att jag har vunnit kunskap om båda.

### Återkoppling:

Jag vet inte om att få feedback har hjälpt mitt eget. Jag tycker dock att det har varit värdefullt för att få andra perspektiv och åsikter. Även om jag inte använt all feedback så ifrågasätter den artefakter man gör och om det finns bättre sätt att göra det på eller varför man gör på ett specifikt sätt. Det är också väldigt inspirerande att se vad andra gör.

Fråga: Det finns objekt som hänger i luften, har ni något förslag på hur jag kan lösa det?

### Feedback:

Under seminariet bad jag om feedback gällande hur man skulle kunna vidareutveckla min artefakt. Jag hade två problem att prefabs placeras på sätt som inte är naturlig och att stigarna genereras lokalt och inte baserat på världens koordinater. Vi diskuterade lite om hur man skulle kunna lösa dessa problem men kom inte riktigt fram till någon definitiv lösning. Vi var dock överens om att lösa dessa problem skulle vara ett bra sätt att vidareutveckla artefakten.

## Reflektion

Jag tycker att jag har uppnått de målen jag satte upp både för mitt eget lärande och med artefakten. De första målen jag satte upp var för en annan artefakt än den jag skapade i slutändan. Jag valde att byta för att jag tyckte att det jag valde att göra var för svårt till att

börja med. Det kändes som att jag inte visste vart jag skulle börja och jag kunde inte komma på ett genomförbart sätt för mig. Dessutom kände jag att artefakten blev för fokuserad på programmerings aspekten och inte tillräckligt mycket på leveldesign. Av dessa två anledningar valde jag att byta målen med min artefakt till en procedurellt genererade skog. Detta kändes enklare och mer genomförbart samt fokuserade mer på leveldesign.

Att jag byte artefakt tror jag bidrog till att jag lärde mig mer både om procedurell generering men också programmering och Unity. Det fanns med dokumenterat om procedurellt genererade skog vilket gjorde att jag lättare kunde skapa mig en uppfattning kring hur jag ville skapa mitt verktyg. Att det fanns mer dokumenterat gjorde att jag lärde mig mer om olika sätt att generera en skog.

Valet av att jobba mycket praktiskt och skapa en artefakt gjorde att jag ökade min kunskap om Unity och Programmering. För att lära mig mer om leveldesign har jag läst en del och har idéer om vad jag vill göra. Jag vill tex implementera stigar för att hjälpa spelaren navigera till viktiga positioner. Jag hann dock inte implementera det då jag inte fick koden att fungera så som jag ville i tid. Jag kunde nog ha valt enklare metoder för att implementera stigar som hade gjort att jag fått in dem, men jag hade en metod jag vill skapa för att minska arbete.

Mitt hållbara perspektiv har ju varit att det förekommer en hel del crunch i spelutvecklingen. Jag tror att bland annat procedurell generering kan hjälpa förebygga crunch, jobba smartare inte hårdare. Om man kan minska arbetsbördan genom att använda verktyg kan man minska utslitning av arbetarna. Detta är ju inte det enda man behöver göra för att lösa crunch. Då redan i industrin finns en förväntar sig att man ska jobba över 40 timmar i veckan. Så attityden hos många behöver även ändras men man måste börja någonstans för att uppnå ett hållbart sätt att utveckla spel.

Jag tycker att min artefakt uppnår hållbarhetsmålen då man kan placera många snabbt och smidigt. Det krävs då inte mycket för att skapa varierade miljöer utan att det tar mycket tid, som då kan läggas på andra områden och minska crunch.

Elin Hallne