# Overzicht lessen SKIL Gamedevelopment

P1

Week 1

Herhaling : Unity Basics, Animator, Builds maken

Week 2

Gitflow

Week 3

Renderpipeline, Structuur & Naming conventions

Week 4

2d Assets , sprites, sprite animations, particles

Week 5

Playability , replayability intorductie

P2

Week 1 valt SD2B uit voor skill!!

Les 1 dus doen in week 2

Week 2 Les 1

Versiebeheer conventies

* Gitflow en Pull requests

Usertesting

Les 2

* Testplan

Les 3

* Test opnames

Les 4

* Analyse en conclusie

Les 5

Visual Sheets (screen to gif + photoshop)

# Lesrooster aandachtspunten

SD2A les donderdag blok 2:

* +-20 studenten
* BWP en ENG gehad in blok 1 zorgen voor goede Energizer

SD2B les donderdag blok 4:

* Laatste blok van de dag tot 18:00
* Studenten laat gestart in blok 3 met 2x BWP
* Letten op verzuimers en te laat komers
* Zorgen dat de studenten thuis zelfstandig aan de gang kunnen met de stof
* +-20 studenten

# Lesplan les 1 : Herhaling : Unity Basics leerjaar 1

**Doel:**

De studenten kunnen Unity goed gebruiken om een scene te bouwen met gameobjecten

De studenten kunnen een simpele mechanic scripten verdeeld over 1 of meerdere scripts

De studenten kunnen animaties toevoegen en aansturen vanuit hun script

De studenten kunnen een simpele UI maken en daarin dynamische data tonen

De studenten kunnen van hun project een build maken.

De studenten krijgen een goed beeld hoe veel tijd ze zelf nodig hebben om een simpele mechanic te bouwen.

**Situatie:**

De studenten hebben deze onderwerpen vorig schooljaar al behandeld in de PROG lessen. Echter zal bij velen deze informatie niet goed zijn blijven hangen..

Enerzijds is dit dus een test en anderzijds een manier om studenten te triggeren om ontbrekende kennis op te halen en te onderzoeken.

**Toetsing:**

De les word een individuele praktijkopdracht die ze in 2 uur af moeten ronden. Eventueel kunnen studenten de opdracht thuis afmaken.

**Vervolgacties:**

Nabespreken van tijdsinschatting versus de realiteit

Nabespreken evt knelpunten in kennis van Unity of programmeren

In kaart brengen en evalueren van de stof uit leerjaar 1

# Lesplan les 1 : Herhaling : Unity Basics leerjaar 1

|  |  |  |
| --- | --- | --- |
| tijd | activiteit | materiaal |
| 00:05 – 00:10 | Landen/Energizer | Nog bedenken |
| 00:10 – 00:15 | Orientatie vragen stellen: Wat heb je vorig jaar geleerd over het bouwen van games? |  |
| 00:15 – 00:20 | Uitleg geven over het belang van de opdracht en de doelen bespreken. | Miro maken voor de les |
| 00:20 – 00:50 | Opdracht uitwerken in tweetallen: Mixamo opdracht | Opdracht mixamo via readme > simulise |
| 00:50 – 01:00 | Bekijken van resultaten | 3 tweetallen studenten laten hun uitwerking zien |
| 01:00 – 01:05 | Doelen bespreken van opdracht | Miro er weer bij pakken |
| 01:05 -01:55 | Opdracht mechanic bouwen individueel uitwerken | Opdracht mixamo via readme > simulise |
| 01:55 – 02:00 | Inpakken en meegeven opdracht als huiswerk |  |

# Lesplan les 2 : Gitflow

**Doel:**

De studenten hebben inzicht in hun persoonlijke tijdsinschatting en of deze klopte (opdracht week 1)

De studenten begrijpen dat branching een belangrijk onderdeel is van een goede versiebeheer

De studeten begrijpen dat voor een goede branching strategie duidelijke afspraken kunnen helpen

De studenten zijn bekend met de “gitflow” strategie

De studenten begrijpen dat ze van de strategie af kunnen wijken en dat het belangrijkste is dat je je aan je eigen afspraken houd

**Situatie:**

De studenten hebben in de week ervoor gewerkt aan een opdracht waarbij ze alle kennis uit leerjaar 1 weer moesten demonstreren d.m.v een goed werkende mechanic in unity. Er zullen dus studenten met hun neus op de feiten zijn gedrukt wat betreft eventuele achterstanden. Het is dus van belang om dit goed in kaart te krijgen en deze studenten ook te activeren om de ontbrekende kennis op te pakken.

Daarnaast hebben de studenten in het laatste project Mythe ervaring opgedaan in samenwerken via git en github en het gebruik van verschillende branches. Dit heeft toen vaak genoeg problemen opgeleverd omdat er geen spraken was van duidelijke afspraken of een goede strategie. Dit kan dus even worden opgehaald om het belang van de les te versterken.

**Toetsing:**

De studenten voeren een opdracht waarbij ze de gitflow strategie moeten doorlopen icm een verschillende simpele opdrachtjes in unity.

**Vervolgacties:**

Studenten met achterstanden moeten extra aandacht krijgen.. Studenten met een achterstand vullen een plannings formulier in over hun eigen plan om de achterstand weg te werken.

De studenten moeten voor hun towerdefence project de “gitflow” strategie gaan toepassen.

# Lesplan les 2 : Gitflow

|  |  |  |
| --- | --- | --- |
| tijd | activiteit | materiaal |
| 00:05 – 00:15 | Nabespreken vorige les: tijdsinschatting en basiskennis, planningsformulier laten zien. En als huiswerk in laten vullen. | Planningsformulier ivm achterstanden in simulize |
| 00:15 – 00:20 | Energizer | Nog bedenken |
| 00:20 – 00:25 | Orientatie vragen stellen: Welke problemen zijn jullie vorig jaar tegegekomen met git? Hadden jullie duidelijke afspraken? Zouden goede afspraken over git helpen om problemen te voorkomen? |  |
| 00:25 – 00:45 | Klassiekale Uitleg over gitflow dmv Miro | Miro (updaten ivm gitflow tool niet meer gebruiken) |
| 00:45 – 01:20 | Opdracht in 3 tallen waarin ze verschillende mini “features” moeten maken en deze volgens de gitflow strategie verwqerken in hun repo | Opdracht gitflow oefening minifeatures op simulise |
| 01:20 – 01:30 | Nabespreken opdracht. Ging het goed? Is het gelukt? Waren er problemen? Welke onderdelen waren lastig? Welke type branches hebben jullie nu gebruikt? |  |
| 01:30 – 01:50 | Studenten verwerken gitflow in hun towerdefense project. Dmv invullen gitflow formulier | Gitflow formulier |
| 01:50 – 02:00 | inpakken |  |

# Lesplan les 3 : Renderpipeline, Structuur en naming conventions

**Doel:**

De studenten weten welke verschillende renderpipelines ze kunnen kiezen in Unity

De studenten weten wanneer ze kiezen voor URP of HDRP

De studenten begrijpen waarom een ordelijke structuur belangrijk is

De studenten weten hoe ze zelf hun structuur in kaart kunnen gaan brengen

**Situatie:**

De studenten hebben in leerjaar 1 tijdens Mythe er varen dat hun Unity project, zeker bij het samenwerken met meer mensen, al heel snel een rommeltje kan worden.

Dit is een mooie reden om het belang van de les aan op te hangen.

**Toetsing:**

De studenten voeren een opdracht uit waarbij ze een structuur opzetten voor hun eigen towerdefense project.

**Vervolgacties:**

In een later stadium (week 6) moet ik de opzet van de structuren weer met de studenten doornemen om te kijken wat er aan veranderd is en waarom?

# Lesplan les 3 : Renderpipeline, Structuur en naming conventions

|  |  |  |  |
| --- | --- | --- | --- |
| tijd |  | activiteit | materiaal |
| 00:05 – 00:10 |  | Energizer | Nog bedenken |
| 00:10 – 00:20 |  | Orientatie vragen stellen: Wie had er vorig jaar een net en opgeruimd unity project ? Konden jullie alle onderdelen makkelijk terugvinden? Wist iedereen waar wat moest komen te staan?  Welke renderpipeline hebben jullie gekozen? Wie heeft er tijdens de productie moeten wisselen van pipeline? |  |
| 00:20 – 00:40 |  | Klassiekale UitlegRender pipelines en Structuur | Google drive [presentatie Alex](https://docs.google.com/presentation/d/1xkeSyOFCSpiBFcU6N0mmRy5CGLrmdWfRQeuW33PWtAI/edit?usp=sharing) |
| 00:40 – 01:20 |  | Opdracht individueel waarin ze hun beoogede unity structuur en naming conventions in kaart brengen via miro | Opdracht [Miro](https://miro.com/app/board/uXjVKyG4PuA=/?share_link_id=875559619958) uitwerken structuur op simulise zetten |
| 01:20 – 01:50 |  | 5 studenten laten op de beamer hun structuur zien en onderbouwen de gemaakte keuzes |  |
| 01:50 – 02:00 |  | inpakken |  |

# Lesplan les 4 : Sprites, Animations & Spritesheets

**Doel:**

De studenten weten wat sprites zijn

De studenten kennen meerdere manieren om in unity 2d nimaties te maken

De studenten kunnen een animatie maken in unity met verschillende onderdelen als sprites

De studenten kunnen een sprite sequence importeren en als animatie implementeren

De studenten kunnen een sprite sheet importeren en implementeren als animatie (particles)

**Situatie:**

De studenten hebben tot nog toe alleen 3d character animaties vanuit mixamo geimplementeerd. Dit is de eerste keer dat we ook de inplementatie van 2d aniamaties behandelen.

Waarschijnlijk gaat Suryani 1 uur in de week bij de flex les met de studenten aan de gang om in aseprite 2d animaties te maken. Zodoende kunnen de studenten dus ook hun eigen animaties gebruiken voor de opdrachten.

**Toetsing:**

De studenten voeren 3 opdrachten uit waarbij ze de verschillende manieren van implementatie van 2d animaties zelf uitvoeren.

**Vervolgacties:**

In het towerdefense project verwerken de studenten zelf ook 2d animaties

# Lesplan les 4 : Sprites, Animations & Spritesheets

|  |  |  |  |
| --- | --- | --- | --- |
| tijd |  | activiteit | materiaal |
| 00:05 – 00:10 |  | Energizer | Nog bedenken |
| 00:10 – 00:20 |  | Orientatie vragen stellen: Wie heeft er in Mythe een 2d game gemaakt? Wie heeft er met 2d animaties gewerkt? Hoe was dat? Heb je de animaties op de juiste manier geimplementeerd? Hebben jullie daar problemen mee ondervonden? Was dit makkelijk te begrijpen? Wat is het verschil en de overeenkomst met 3d animaties? |  |
| 00:20 – 00:40 |  | Klassiekale Uitleg 2d animations | Google drive [presentatie Alex](https://docs.google.com/presentation/d/1CL6mbHN3z-tAK34Aa6DXf7J29w2wlj_IaALCrTH31w0/edit?usp=sharing) |
| 00:40 – 01:00 |  | Demo video kijken animeren 2d character in unity. Evt mogen de studenten proberen zelf mee te doen en te reproduceren. Meegeven als huiswerk. Op simulise | Demo video met een 2d character idle , walk en attack animation |
| 01:00 – 01:30 |  | Opdrachten in voor het implementeren van een sprite sequence (explosion) en een spritesheet (partice effect). Gebruiken effecten bijvoorbeeld via <https://itch.io/game-assets/tag-effects>  Evt eigen sequences of spritesheets uit aseprite? Suryani flex? | Opdrachten op simulise (readme) met stap voor stap instructie |
| 01:30 – 01:50 |  | Studenten laten zien wat ze hebben kunnen maken op de beamer. Zijn ze trots? |  |
|  |  |  |  |

# Lesplan les 5 : Usertesen, playability, replayability

**Doel:**

De studenten weten het verschil tussen playtesten en usertesten

De studenten weten aan welke eisen een usertest moet doen

De studenten kunnen een testplan maken

De studenten kunnen een doelgroep bepalen

De studenten weten wat playability en replayability is en kunnen bedenken waar ze op kunnen letten om hierop te testen

De studenten kunnen een usertest uitvoeren en in beeld brengen

**Situatie:**

Tot nu toe zijn de studenten vooral bezig geweest met het uberhaupt werkend krijgen van mechanics voor hun games. Er is nog nooit echt tijd en aandacht geweest voor het kijken naar de kwaliteit van wat er gemaakt is.

Het is nu van belang dat de studenten bewust worden van kwaliteitseisen en hoe deze voor games via usertesting beoordeeld kunnen worden.

**Toetsing:**

De studenten schrijven een testplan voor het testen van een game die ze op hun eigen steam account hebben staan. Ze bepalen de doelgroep van de game en gaan met deze doelgroep een usertest houden volgens plan. Deze nemen ze op en gebruiken ze om een analyse uit te voeren en uiteindelijk conclusies te onderbouwen.

**Vervolgacties:**

In de volgende les (week) presenteren de studenten hun usertests

# Lesplan les 5 : Usertesen, playability, replayability

|  |  |  |  |
| --- | --- | --- | --- |
| tijd |  | activiteit | materiaal |
| 00:05 – 00:10 |  | Energizer | Nog bedenken |
| 00:10 – 00:20 |  | Orientatie vragen stellen: Wie heeft er wel eens games getest? Wat is testen? Wat test je dan? Hoe bepaal je of iets goed is? Waar kijk je naar? Wat gebeurt er met de resultaten van een test? Wat is het doel van testen? |  |
| 00:20 – 00:40 |  | Usertesten , playability , replayability | Miro presentatie maken over playability replayability en usertesten |
| 00:40 – 01:00 |  | Bekijken usertest footage examens. Wat vinden de studenten ervan? | Usertest footage voorbeelden |
| 01:00 – 01:30 |  | Opdracht voor het schrijven van een testplan 2 tallen. Inleveren op simulize | Opdracht op simulise testplan |
| 01:30 – 01:50 |  | Bespreken of iedereen volgens plan kan gaan testen en opnemen? Bespreken deadline voor inleveren testfootage op simulize | Opdracht footage op simulize |
| 01:50- 02:00 |  | Inpakken + delen presentatie schema volgende week | Presentatie schema |

# Lesplan les 5 : Usertesen, playability, replayability

**Doel:**

De studenten weten het verschil tussen playtesten en usertesten

De studenten weten aan welke eisen een usertest moet doen

De studenten kunnen een testplan maken

De studenten kunnen een doelgroep bepalen

De studenten weten wat playability en replayability is en kunnen bedenken waar ze op kunnen letten om hierop te testen

De studenten kunnen een usertest uitvoeren en in beeld brengen

**Situatie:**

Als het goed is hebben de studenten geusertest en dit verwerkt tot goede usertest footage, deze geanalyseerd en hierover conclusies getrokken.

De studenten presenteren deze les hun testfootage en hun bijhorende analyse en conclusies.

**Toetsing:**

De studenten presenteren hun usertests, analyse en conclusies

**Vervolgacties:**

Studenten gaan ook hun eigen game nog usertesten voor de BO

# Lesplan les 5 : Usertesen, playability, replayability

|  |  |  |  |
| --- | --- | --- | --- |
| tijd |  | activiteit | materiaal |
| 00:05 – 00:10 |  | Klaarzetten presentaties + doornemen schema | schema |
| 00:10 – 01:30 |  | Presenteren door elke groep (10x) 8 min per groep. Alle feedback aan het einde… |  |
| 01:30 – 01:50 |  | Feedback op de presentaties: Waren de tests goed uitgevoerd?waren ze goed voorbereid? Was de doelgroep goed? Is er een goede analyse geweest? Zijn de conclusies onderbouwd?Is de footage goed te volgen en te verstaan? |  |
| 01:50- 02:00 |  | inpakken |  |