Procedural Art

Term 2.3

Course Manual study year 2022/2023

Bachelor Creative Media and Game Technologies (CMGT)
School of Creative Technology

Publication date	13 / 02 / 2023
Version	2.1
Module coordinator	Max Klostermann (MKL24)
Lecturers	Max Klostermann (MKL24), Paul Bonsma (PBO06), Mark Schipper (MSC19), Malik Nabil (MNA05), Iain Douglas (IDO05), Simon Buijs (SBU05), Daniel Valente de Macedo (DVA06) and Taco van Loon (TLO02), Bram den Hond (BHO03)
CMGT roles	Artist | Designer | Engineer

1 General overview

Module Name	Procedural Art
Unit code	L.28086
Year and Term	2.3
CMGT roles	Artist, Designer and Engineer
Credits	3 ECTS
Lessons	Labtures: 4 * 7 hours
Study load	84 hours
Responsible lecturer	Max Klostermann (m.m.klostermann@saxion.nl)
Lesson structure	Bootcamps in form of labtures (theory and practical work combined)
Module summary	The student gains theoretical and practical knowledge about different means of generating procedural content, understands how procedural workflows can be utilized to create and enhance visual output, establishes a non-destructive pipeline, accelerates iterations and thus lowers the overall amount of production time in current and future projects. The product delivered is a procedurally crafted and reimagined small-scope city area produced by using suitable tools and workflows, presented in the game engine Unity.
Industry relevance	Procedural workflows are becoming more and more standard for both real-time software applications such as Game engines and offline-rendering tools utilized for Visual FX and CGI in the movie industry. Acquiring a basic understanding on how products and projects can be approached and enhanced procedurally is important in order to have a foundation for critical conversation and exchange with other professional fields related to Creative Media & Game Technologies.
Type of exam	Assessment (based on uploaded products)
Exam code	T.54133
CMGT Competencies	1. Technical research and analysis 2. Designing, prototyping and realizing 3. Testing and rolling out 5. Conceptualising 6. Designing
Required prior knowledge and skills / conditions for enrolment	Engineers: Unity Game Scripting. Recommended: Algorithms. Designers: Unity Game Scripting. Recommended: 3D Modelling. Artists: Unity 3D Art. Recommended: 3D Modelling.
Preparatory for	Project Show-Off and a multitude of other future projects

2 Why this module?

It is generally advisable for future projects that students obtain theoretical knowledge of procedural workflows in order to **generate** digital **content** in a **procedural manner as opposed to investing** too much **time into tedious** and **manual labour**. Understanding and being able to apply this knowledge is important for **establishing** a **pipeline** which **improves the workflow** and lowers the time it takes to author content for everyone involved **in the long run**. We teach this in the context of **small-scale assignment** using the game engine **Unity** as well as pre-existing tools such as **Adobe Substance 3D Designer** as well as self-made tools that enable all CMGT roles to create re-usable, adaptable, and non-destructive procedural content with an **individual approach** and **focus**.

In order to develop media products, both for the movie- and game industry, you need to have knowledge about the current technologies used in the field which will enable these new workflows. We use technologies and approaches to **let digital content be generated for us** - the developer and the user as well. It is important to understand what happens and how things happen in between: how content is authored procedurally, what its benefits are and why we should make use of it as developers of interactive content.

Students need to understand the foundations of procedural art and ask themselves questions like:

- *What caused procedural pipelines in the industry to emerge and why?*
- *What are the possibilities and limitations of procedural workflows?*
- *When does a procedural workflow make sense and when does it not?*
- *What is the current state of procedural workflows in a professional pipeline?*
- *How will procedural workflows evolve and how is that potentially going to look like?*

2.1 What happens in the labs and lectures?

In this year's iteration of the module, you will learn about theory, concepts, look at and work on practical exercises and get to develop scripts and assets for your own experiments and the module's assignment. This year's version of the Procedural Art module will facilitate a departure from previous years as it **won't be scheduled as** and **split up in lectures and labs**. Instead, it will be conducted in form of so-called **bootcamps**. This departure is necessary to test and experiment with the upcoming way of teaching content soon, preparatory to the upcoming changes to CMGT's curriculum and will yield valuable information and data that can be used to improve future iterations of this module in advance.

These **bootcamps** will enable a different way of teaching and leads to a more focused approach on learning about a specific subject. While it used to be taught in form of lectures (theory and concepts) and labs (practical exercises), it is now converted into so-called **labtures** that will merge lectures and labs into one combined type of structure.

The **labtures** for each class you will participate in will partially vary in content. The engineering classes will focus on more scripting-focused contents, working with theory, concepts and practical exercises utilising scripting and producing procedural content through code using C# in the Unity game engine. The designers and artists will participate in **labtures** that will mainly tackle procedural material creation using Substance 3D Designer and learn how to incorporate these assets by combining them with modular assets to create 3D environments in Unity, making these labtures more **design- and art-focused**. Students are **free to attend either** set of **labtures** each week if their schedule and planning allows for it. Furthermore, pre-recorded lecture material from the previous year will be available for those who won't switch classes at any point, making all content accessible for all classes. Chances on missing out on information are greatly limited that way, which we consider a big plus.

The structure of labtures will for the most part focus on teaching concepts, discussing theory, and showcasing workflows switched up by introducing sections of time in which students will do **research**, work on **scene** & **asset creation**, **tooling**, **experiments,** and **lab assignments** by using the tools and processes necessary for finishing them. Please make sure to check the **expertise** of **each lab teacher** in case you have a **specific question or problem** in a particular field of expertise the current lab teachers do not offer. This might mean that you need to find alternative ways of guidance, as in e-mailing or visiting other labtures with a different focus.

We will work a lot with **Unity**, an **IDE** of your choosing and a software package aimed towards a node-based, procedural workflow, namely **Substance 3D Designer** from **Adobe**. To get an understanding on how a procedural pipeline is set up, you will learn how to create and customize digital content on an intermediate level.

2.2 How does this module relate to other modules in the CMGT study program?

We expect that both engineers and designers followed the **Unity Scripting module**, so that you are able to work with the Unity game engine that we will use during both lectures and lab classes. Being able to apply knowledge obtained during the **Algorithms module** is a big plus. The artists are expected to have followed at least the **Unity3D Art module**. Knowledge and skills obtained during the previous **3D Modelling course** will be beneficial for both designers and artists. Besides that, this module teaches you the basics of acquiring a ‘procedural mindset’ which you can and will apply in future projects and work in general. It prepares for many of the following modules, be it a subsequent module or group project since procedural workflows are universally applicable to most digital projects.

3 What are you going to learn in this module (learning objectives)?

The student:

1. **acquires** a procedural **mindset** in being able to **create** digital content **non-destructively**.
2. **understands** the **structure** of **meshes** and procedural **materials**.
3. **understands** the **modular assets workflow** for procedural scene creation.
4. **is able to** generate **reusable** and highly **adaptable content** for a multitude of projects.
5. **knows** about different **physically based** (PBR) **materials**, **textures** and **shaders**.
6. **understands** methods and techniques for **optimizing** digital **content** for a **real-time** application.

4 Which resources do you need?

Any resources, recommendations, links to subsequent information and files will be conveyed during the labtures and made available on Blackboard afterwards. Furthermore, the student needs to obtain a free **Unity** license (store.unity.com/ - plans-individual). We recommend using a **Unity LTS version (Long Term Support)** or **Unity 2021.3.18** as well as its render pipeline **HDRP** (High-Definition Render Pipeline) **template preset**. For creating procedural materials, students will want to install a free student license from **Adobe**’s node-based procedural software package called **Substance 3D Designer** (adobe.com/products/substance.html), **NOT Adobe Substance 3D Painter** (since Painter exports bitmap textures only) - the only **exception** for using *3D Painter* in this module being the creation of additional prop textures (see rubrics). Next to this, please **include Microsoft Visual Studio** in your **Unity installation** (can also be done after the fact in your **Unity Hub**: *Installs -> Add components,* as this allows for code completion in Unity). Alternatively, please download and install an **IDE** of your choosing. **Microsoft Visual Studio** is recommended in any case (visualstudio.microsoft.com/). Additional Unity assets such as the **‘Substance in Unity’ plugin** can be downloaded from the Unity Asset Store for free (https://assetstore.unity.com/packages/tools/utilities/substance-3d-for-unity-213208). Try whether they work with your installed Unity version (if it doesn’t, ignore the substance implementation into Unity and showcase the procedural capabilities of your substances outside of the engine eventually).

5 What does the programme of this module look like?

On Blackboard you'll find the course content and a detailed course overview.

Bootcamp	Lecture/Lab	Topic(s)
1	**Labture**	**Engineers**: Introduction to procedural workflows, modular meshes and shape grammars

1	**Labture**	**Designers/Artists**: Procedural Art primer, Industry methods, Basic material creation
2	**Labture**	**Engineers**: More on modular meshes, creating Unity editor tooling
2	**Labture**	**Designers/Artists**: Procedural Material primer, implementation, material exercises
3	**Labture**	**Engineers**: Basic mesh generation in Unity
3	**Labture**	**Designers/Artists**: Modular Assets primer and theory, Unity scene implementation
4	**Labture**	**Engineers**: Advanced mesh generation in Unity, themes reveal
4	**Labture**	**Designers/Artists**: Tips for the assignment, Grid City script, themes reveal, research

6 How is this module assessed?

6.1 Assessment

The assessment encompasses the theory from the labtures, experiments conducted during those, and the subsequent results worked on and polished throughout the remainder of the time in form of a final product. The assessment will be done based on a submission on Blackboard and the evaluation by the teachers involved.

There are two assessments:

- the first in week 3.9 where you need to submit the Unity project, the research document, assets, tools & building blocks and a video
- the second opportunity will be at the end of next term in week 4.9 (redo)

For the final submission, each student will have to present their final product in a concise manner by means of recording a video. A presentation template will be provided which includes slides that merely need to be filled with individual content made by the student and should cover all important aspects of the video and what contents are expected to be presented.

Furthermore, we will track your progress on a regular basis during the labtures for everyone who needs guidance and help.

See the rubric of the assessment on the last page.

6.2 Procedure

In order to be able/be allowed to participate in this course, you should enrol yourself for the course on Blackboard.

6.3 Criteria & Assessment form

The assignment is to create a reimagined small-scale city area from one of four pre-selected video game cities/areas, or a self-chosen existing video game city/area of similar complexity and visual fidelity. Students will work with the game engine Unity to author a three-dimensional, small, and procedurally crafted area of either a city block, a village or even a hamlet based on the chosen theme. The four suggested themes will be revealed during the final labture.

Deadline: The final submission needs to be handed in on Blackboard on Sunday, the **23rd of April 2023**, 23:59h at the latest.

All criteria of this assignment will be analysed and graded using rubrics. If the evaluation shows that certain criteria hasn't been met, those aspects will be graded with an insufficient. In order to pass the course, the score of your submission needs to be equal or higher than 9 points (see rubric below). In case there are any adjustments, it will be communicated through Blackboard in due time.

6.3.1 What do you have to hand in and how?

Deliverables (preconditions):

- A **Unity project including** a **Unity scene** that is at least partially procedurally generated (either in **editor**, **or** in **play mode** without compiling errors) and **represents** a **reimagined city/village area** based on **one** of the recommended four **themes** or **a self-chosen video game city/village of similar complexity** and **visual fidelity** (upon consultation with your teacher)
- The created **assets, tools** and **building blocks** (can be Substances, C# scripts, Shaders, 3D meshes etc.)
- A **research document** based on your chosen theme, including visual research, additional research into relevant procedural techniques, potential use of external assets, contributions, and sources (**5 pages max!)**
- A **video** showing all results, demonstrating their procedural tooling/capabilities, a breakdown of and highlights in scripts/assets/materials and showcasing the resulting small-scape city/village area in Unity (including an overview of the research you've done; this should be an overview of your research document)
- Upload all files to the respective submission categories as **.zip** or .**rar** file
  - file size limit for Unity project: 1000 MB (1 GB)
  - file size limit for Tools & Building blooks: 500 MB
  - file size limit for Research document: 50 MB
  - file size limit for Video: 300 MB
- Name your files **FULLNAME_STUDENTNUMBER_CMGT-ROLE_THEME_'UNITYPROJECT' or 'ASSETS&TOOLS&BB or 'RESEARCHDOC' or 'VIDEO' .zip/.rar** respectively

Note: please do not include the Library, obj, Logs and .vs folders (and possibly .git folder) in your Unity project submission.

Important note about plagiarism: You may (and probably should) use the Handouts and Substances found on Blackboard to learn from, and you may use free external assets (such as Kenney's modular meshes) for the final assignment, if you clearly indicate what your own contribution is, as part of your research document at the very least. You will only be assessed and graded on your contribution. You may even use scripts/meshes/materials from other students under the same condition (see 'Rules for distribution' below for more additional information). Note that making a minor change to an asset from another student or from the internet, and presenting it as your own, is plagiarism. This will result in a 1 as a grade, and possibly additional measures. If you have shared your assets with other students and/or have used assets from someone, please also indicate this in your research document and credit their work, to avoid being the subject of these plagiarism measures.

Rules for distribution: The following notion is based on the official CMGT role you have signed up for in the beginning of the CMGT study course (you are free to choose what contents to create for this module regardless): As an artist you are not allowed to use assets from other artists in this module. As an engineer you are not allowed to reuse assets & code from other engineering students in this module. As a designer you are not allowed to use more than one art-related asset and one engineering-related asset from other students in this module. Note that the use of assets has to be mentioned and clarified in your research document as in, you give detailed information about the source and credit the original creator. If you are the original creator of an asset (scripts, art, substances etc.) you are allowed to share your asset with a maximum of 5 students that do not share the same CMGT role with you (artist/designer/engineer).

6.4 Redo

A second opportunity to submit the redo on Blackboard will be at the end of term 4 in week 4.9 on Sunday, the **9th of July 2023**, 23:59h at the latest. Keep an eye out for the announcements for more information.

7 Who are the contact persons for this module?

Contact person	E-Mail	Expertise
Module coordinator		
Max Klostermann	m.m.klostermann@saxion.nl	Substances, Modular Assets, 3D Art/VFX, Optimization, Texturing
Lecturers		
Mark Schipper	w.g.a.schipper@saxion.nl	Modular Assets, Substances, Shaders, Optimization, 3D Art/VFX
Malik Nabil	m.m.nabil@saxion.nl	Modular Assets, Texturing, Optimization, 3D Art/VFX, Substances
Paul Bonsma	p.s.bonsma@saxion.nl	Unity Scripting, Algorithms, Mesh Generation, Code architecture
Simon Buijs	s.d.buijs@saxion.nl	Unity Scripting, Algorithms, Mesh Generation, Code architecture
Iain Douglas	i.d.douglas@saxion.nl	Modular Assets, 3D Art, Prop Texturing
Taco van Loon	t.vanloon@saxion.nl	Modular Assets, Substances, 3D Art, Prop Texturing
Daniel Valente de Macedo	d.valentedemacedo@saxion.nl	Unity Scripting, Algorithms, Mesh Generation, Code architecture
Bram den Hond	a.c.denhond@saxion.nl	Unity Scripting, Algorithms, Mesh Generation, Code architecture

8 Rubric

Note: the rubrics are used to determine your grade and are visible in Blackboard under ‘Grades and Feedback’ --> ‘View rubrics’.

Rubric - Procedural Art 2022/23

Preconditions:

- A **Unity project** including a **Unity scene** that is at least partially procedurally generated (either in editor, or in play mode without compiling errors) and represents a reimagined city/village area based on a chosen existing theme
- The created **assets, tools** and **building blocks** (can be C# scripts, shaders, 3D meshes like fbx or obj, native substance files like sbs, additional textures for props etc.)
- A **research document** based on your chosen theme, including visual research, additional research into relevant procedural techniques, potential use of external assets, contributions, and sources (5 pages max; the research document also contains a clear and truthful overview of what your own contributions are)
- A **video** showing all results, demonstrating their procedural tooling/capabilities, a breakdown of and highlights in scripts/assets/materials and showcasing the resulting small-scape city/village area in Unity (including an overview of the research you’ve done; this should be an overview of your research document)

If you do not meet the preconditions, the submission will be graded with an insufficient. Plagiarism will result in a 1 and potentially also additional measures. The use of Substance Painter for authoring textures is also prohibited with one exception: the creation of textures for additional props (see category 5 in the rubric).

Grading: Divided into five categories, 25 scoring aspects (criteria) have been given (28 points possible in total). You can score 0, 0.5 or 1 points in each aspect (0 points = not satisfied, 0.5 points = satisfied on a basic or mediocre level, 1 point = satisfied completely) with three exceptions where a student is able to score 2 points max. for three particular aspects (optimization, materials/substances and tooling).
Grade = 1 + #points/2 (with a maximum of 10). Note: you only need to score 17 points (1+17/2 = 9.5) in total for a 10 as your final grade.

	Criteria	Not satisfied	Basic / mediocre	Satisfied completely
1. Theme-based Research (4 pts) You analyse several methods of procedural content creation to explore potential solutions and research visual reference.	A clear description of architectural elements is given.	0	0.5	1
	A clear description of the city shape is given (roads, neighbourhoods, terrain elements, top-down-structure).	0	0.5	1
	A clear description of materials is given (necessary amount, state of material, material properties).	0	0.5	1
	New procedural techniques or tools have been researched and described ("new" = not treated in detail during the labtures).	0	0.5	1
2. Meshes, UV and Structure Generation (10 pts) You are able to create mesh structures procedurally and / or make use of modular assets to design buildings with real-time optimization in mind.	A wide range of shapes is created procedurally from a smaller range of building blocks.	0	0.5	1
	Meshes are created or modified procedurally (e.g. lathe, extrude, warp).	0	0.5	1
	The resulting structures match the visual research.	0	0.5	1
	Meshes (modules) are created that fit the style.	0	0.5	1
	A consistent scale for size and placement of buildings and other structures has been established.	0	0.5	1
	There is a lot of recognizable variety in the buildings (i.e., not "bowls of oatmeal" style variety).	0	0.5	1
	Mesh modules are UV'd to enable tiling.	0	0.5	1
	Procedural meshes are textured without extreme artefacts, e.g., stretching, stitches (this can be done with procedural UVs or by shaders).	0	0.5	1
	Optimizations have been done for real-time efficiency (think: LOD groups, draw call optimization, careful observing poly counts, mesh welding).	0	1	2
3. Materials, Shaders and Textures (6 pts) You apply procedurally generated physically based materials and texture types to mesh-based structures in a procedural manner.	An appropriate amount of believable looking High-Quality PBR materials (substances) have been created behave realistically and are efficient in performance.	0	1	2
	A suitable amount of texture variety has been produced that match the visual style and are used in the scene to capture important characteristics & details of the chosen theme.	0	0.5	1
	Good choices have been made regarding texel density and texture sizes.	0	0.5	1
	All textures have the proper scale (as applied in the scene onto their respective surfaces).	0	0.5	1
	Custom shaders are used and enhance the scenery (as in boosting the visual fidelity, e.g. motion, water, raindrops done through Shader Graph or code).	0	0.5	1

4. Customizability and Control (5 pts) You can showcase complex and diverse content creation by means of providing control and tooling methods.	There is custom (Unity editor) tooling for fast scene creation (e.g., building placement, road drawing).	0	1	2
	Buildings can be customized after generation.	0	0.5	1
	Building parameters can be controlled before generation (e.g., different neighbourhoods have different building styles).	0	0.5	1
	Materials can be customized (exposed parameters right inside of Unity).	0	0.5	1
5. Look and Feel (3 pts) You are able to provide polished visual results by creating a plausible, re-imagined adaptation of the chosen theme.	Additional props and an appropriate number of city-specific characteristics were added to increase detail and realism.	0	0.5	1
	Appropriate lighting and Post Processing / VFX have been applied to enhance visual fidelity.	0	0.5	1
	The overall look matches the chosen theme.	0	0.5	1

Total amount: 1+ ____ /2

Grade: ____

Points under **Total amount** (pre-calculation)					
≤ 8.5 points = Insufficient (1 - 5)	9 – 10.5 points = Sufficient (6)	11 – 12.5 points = Decent (7)	13 – 14.5 points = Good (8)	15 – 16.5 points = Great (9)	≥ 17 points = Excellent (10)