

Fig. 01 – Rhino to Nano Banana Interior View



Juliette Zamani Alavijeh
Caterina Lovo
Domina Lučin
Niklas Murmann
Cyrill Haas
Lisa Holbrook

CPLX

Cave Pulse Light Experiment

AR122

1:1 Interactive Architecture Prototypes

27/03/2026

1 LOCATION | DESIGN CHALLENGE

- Antarctica is the coldest, windiest, and driest continent on Earth
- -10°C (Coast) to -60°C (Interior)
- Katabatic winds (Wind speeds can exceed 100 km/h for days at a time)
- Blizzards (Temperature is less than 0°C and visibility is reduced to 100 m or less)

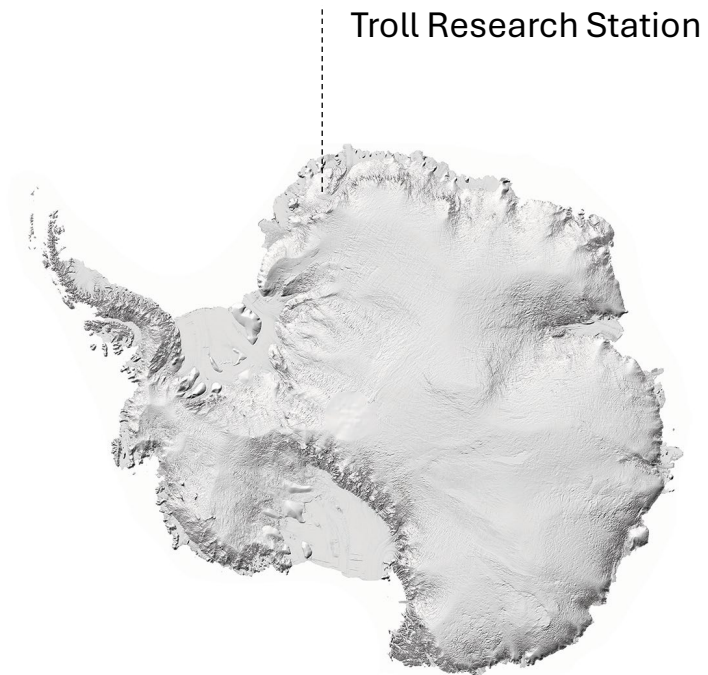


Fig. 03 – Map Terrain Antarctica



1 LOCATION | DESIGN CHALLENGE

- Approx. 50 days of polar daylight in summer (November to January)
- Approx. 60 days of polar night (May to July)
- 63% of expeditioners reported:
 - Difficulty falling asleep
 - Feeling sleepy during the daytime
 - Delayed sleep onset and offset time
 - Reduced sleep duration
 - Decreased sleep efficiency
 - Aggravated sleep fragmentation
 - And altered sleep architecture

Source: Liu et al., 2024, Nature magazine

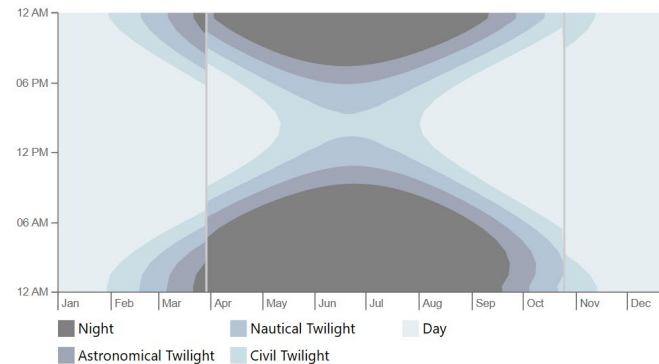
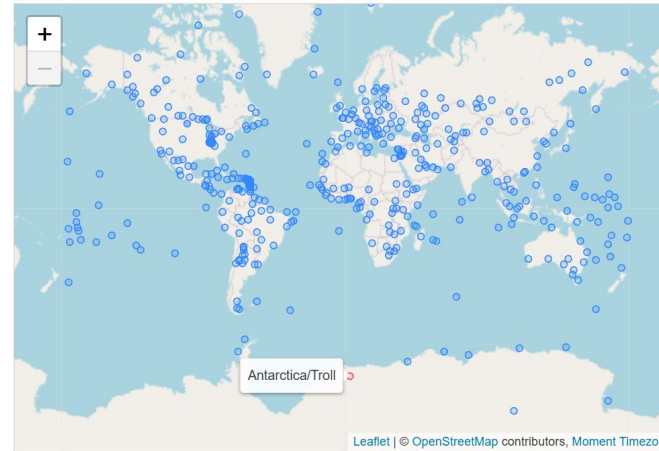
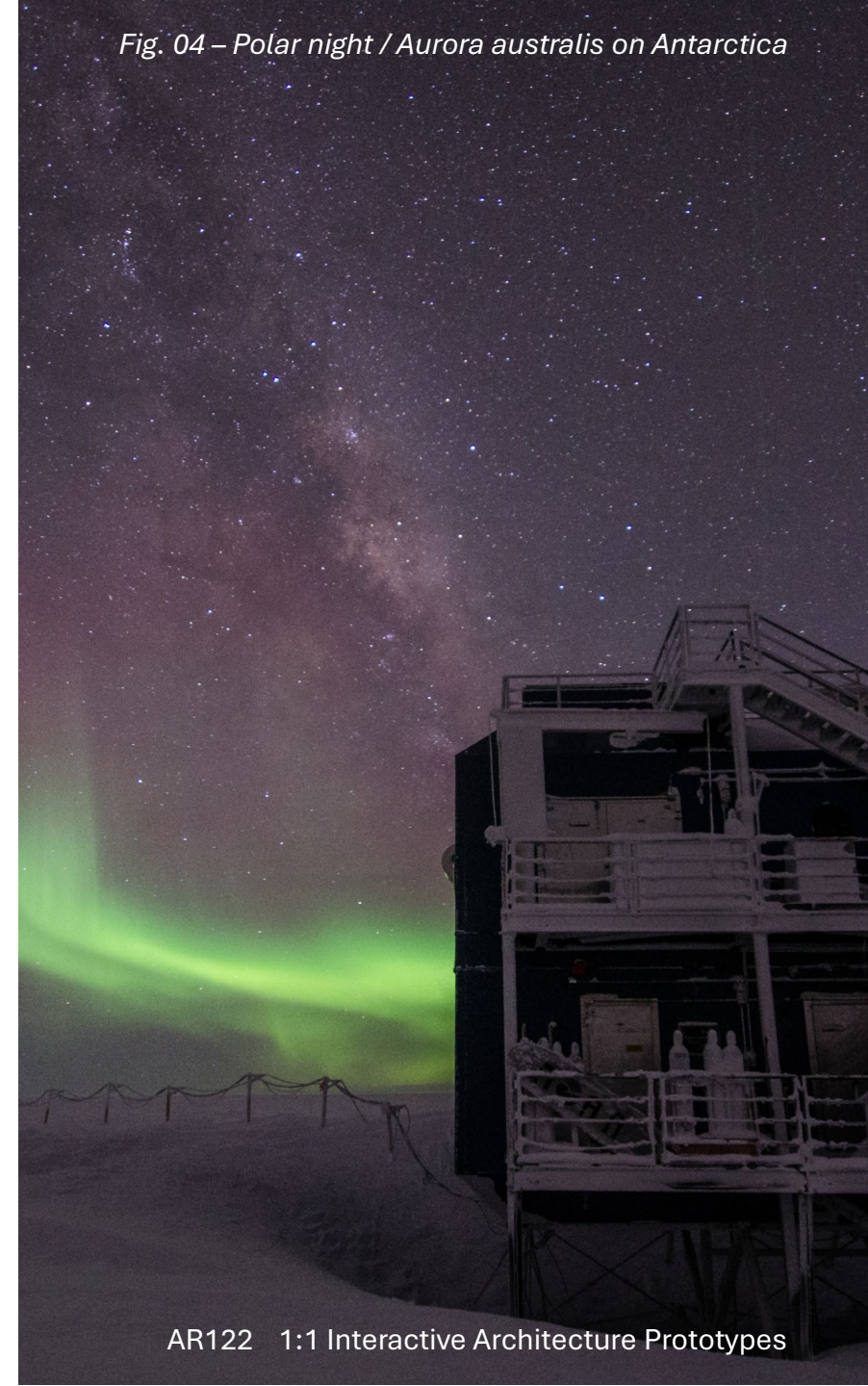


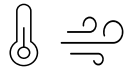
Fig. 05 – Daylight conditions – Troll station



INTRODUCTION

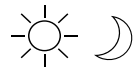
1 INTRODUCTION | PROJECT GOAL

CHALLENGES



Weather conditions

- Severe Temperatures
- Strong Catabatic Winds
- Minimal Precipitation
- Low Humidity



Arctic seasons

- Extreme Light Cycles



Lighting conditions

- Circadian Rhythm Disruption
- Reduced Work Efficiency
- Sleep & Mood issues



Spatial limitation

- Lack of Privacy
- Lower Productivity
- Stress and Discomfort
- Design compromises

IMPACT ON PHYSICAL AND MENTAL HEALTH

Fig. 07 – Functional division



1 INTRODUCTION | **FUNDAMENTAL NEEDS**

Autonomy	Impact	Stimulation
Beauty	Morality	Security
Comfort	Purpose	Fitness
Community	Recognition	
Competence	Relatedness	

Fig. 08 – Fundamental needs



1 INTRODUCTION | **FUNDAMENTAL NEEDS**

Autonomy	Impact	Stimulation
Beauty	Morality	Security
Comfort	Purpose	Fitness
Community	Recognition	
Competence	Relatedness	

Fig. 08 – Fundamental needs



1 INTRODUCTION | PROJECT GOAL

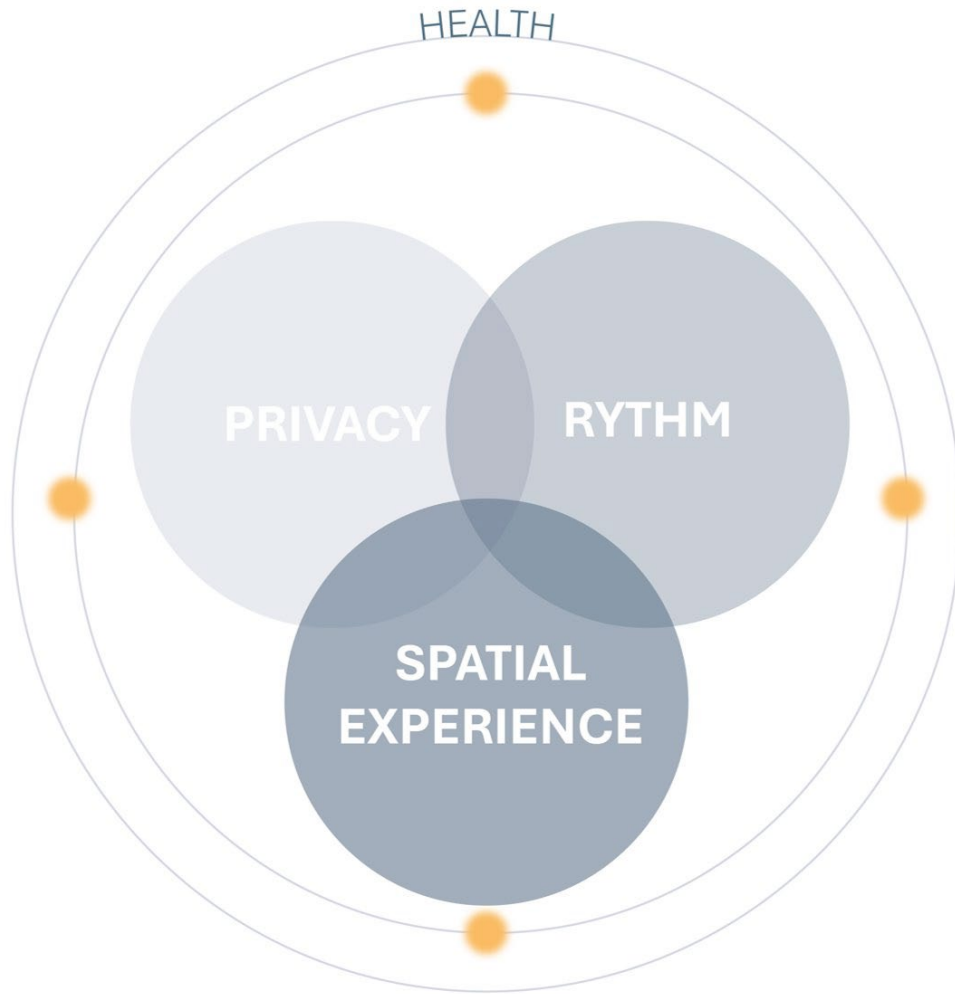


Fig. 09 – Project goals



1 INTRODUCTION | PROJECT GOAL

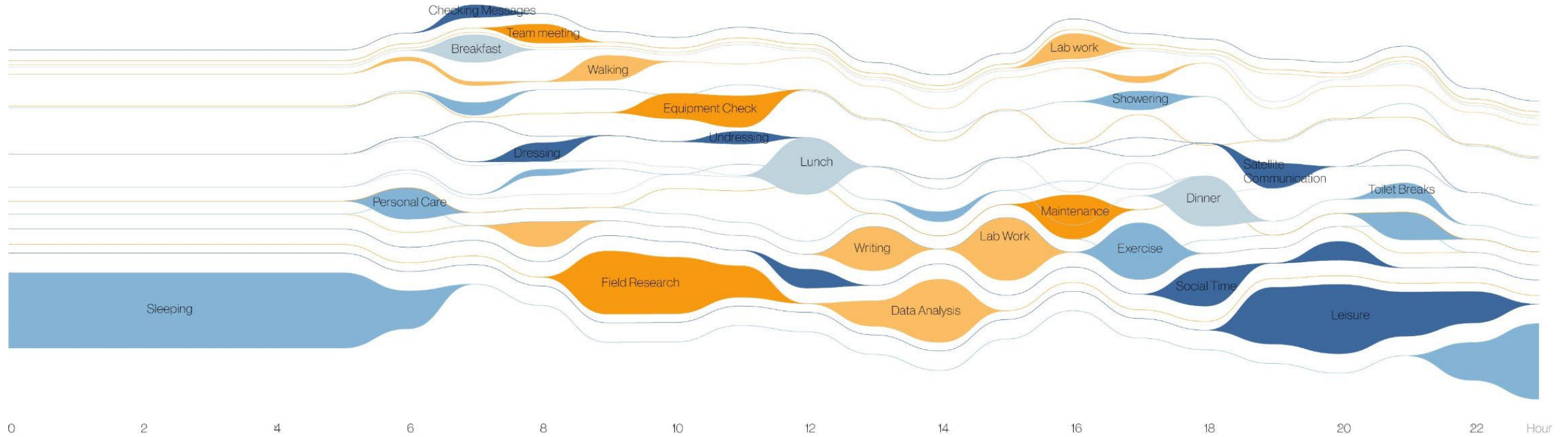


Fig. 10 – 24h activity mapping

DESIGN CONCEPT

2 DESIGN CONCEPT | THE CAVE EXPLORATION



Fig. 11 – Concept: Cave



2 DESIGN CONCEPT | STACKING

- Spatial Variety
- Amplify the quality of each function
- Vertical corridor
- Reduced footprint
- Less foundation

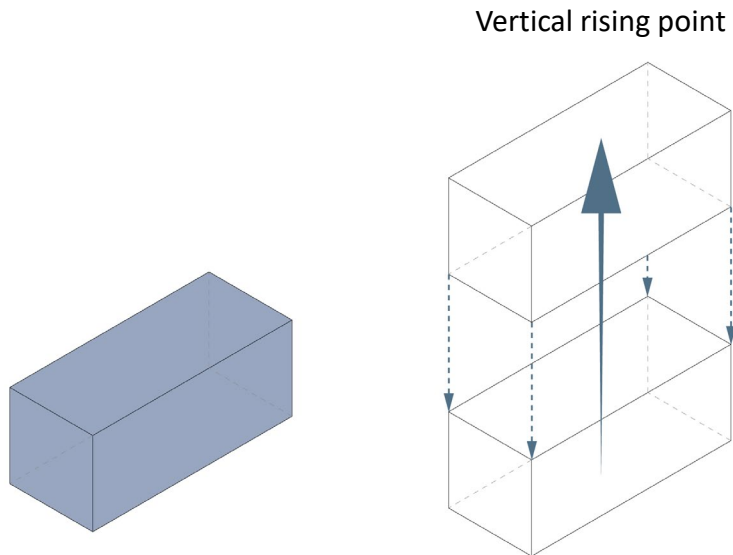


Fig. 14 – Diagram stacking



2 DESIGN CONCEPT | ARRAYING

- Air Lock as a horizontal connector

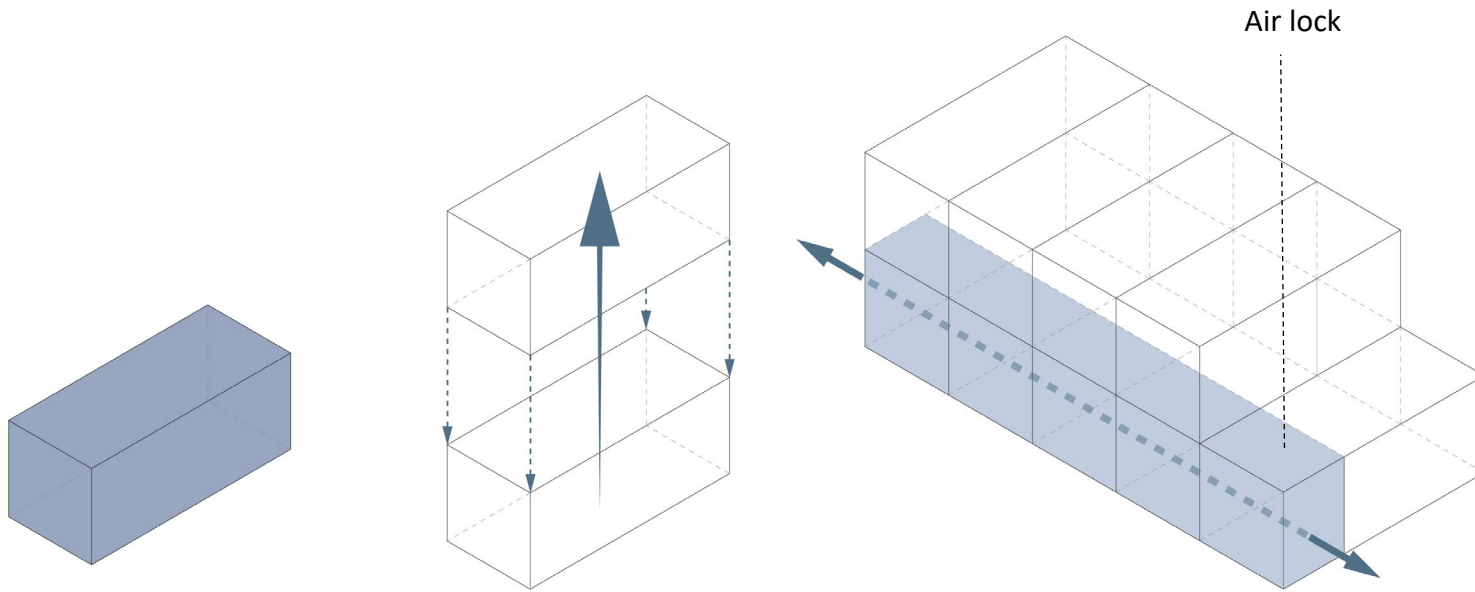


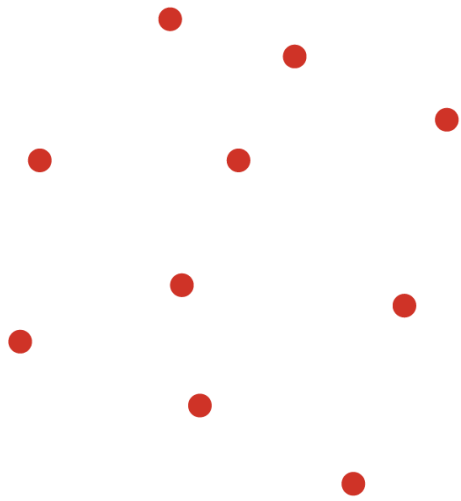
Fig. 14 – Diagram stacking & arraying

Fig. 12 – Exterior Vision, AI generated (Nano Banana)

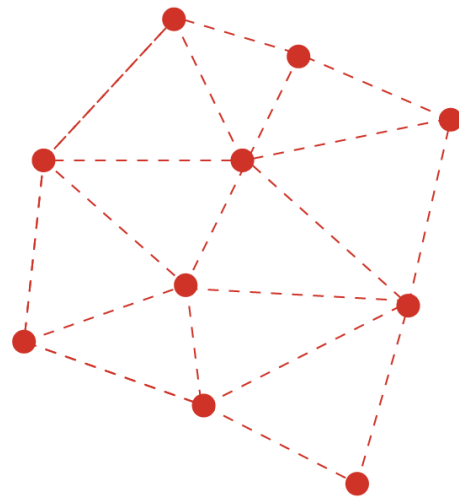


METHODS

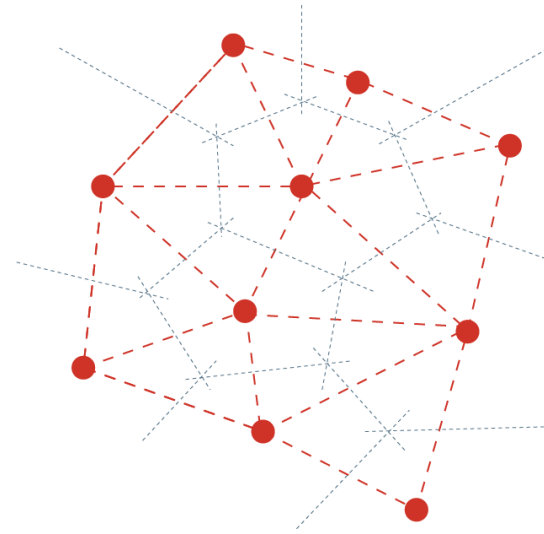
3 METHODS | VORONOI



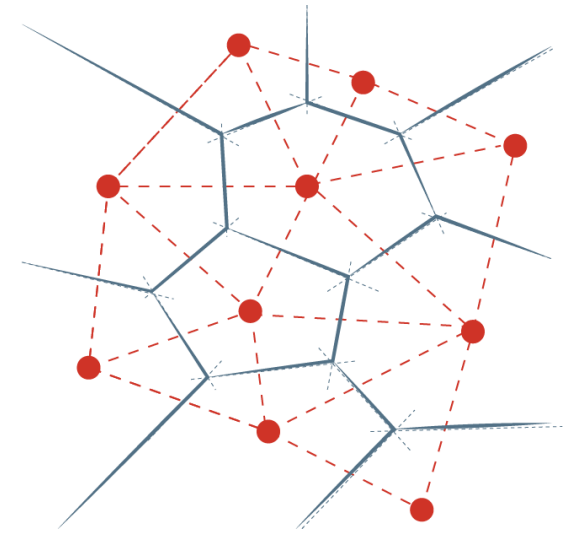
1) Seed distribution within the design domain (adaptive node definition)



2) Delaunay triangulation



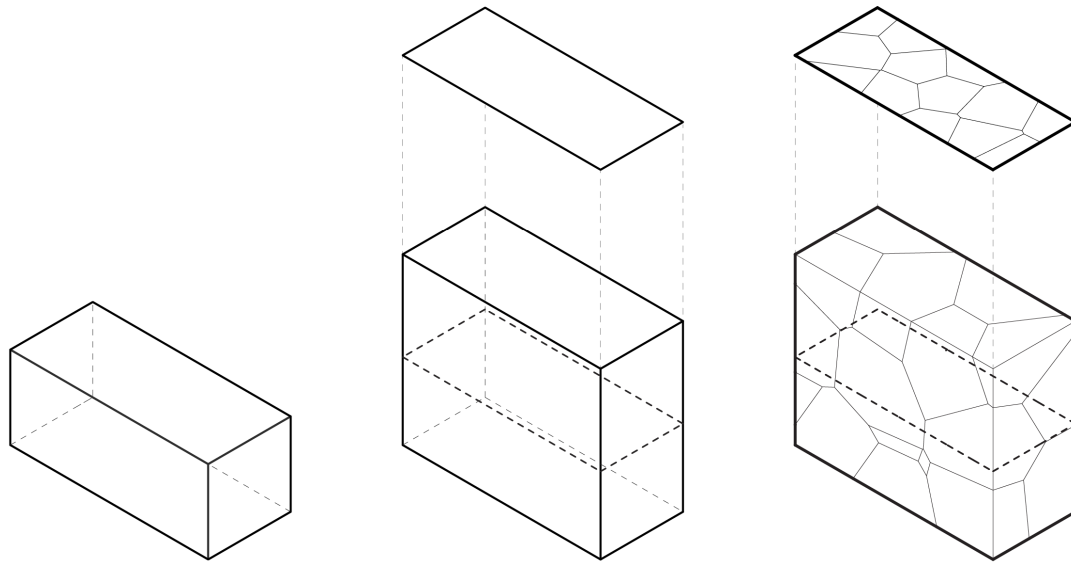
3) Bisector line identification



4) Voronoi tessellation - (2d polygons)

Fig. 13 – Voronoi principle

3 METHODS | TOOLBOX



Create container

**Stack and
add floor**

**Apply voronoi
for zoning**

Fig. 14 – Diagram form finding script

Group 3

Fig. 12 – Exterior Vision, AI generated (Nano Banana)



AR122 1:1 Interactive Architecture Prototypes
AR122 1:1 Interactive Architecture Prototypes

3 METHODS | AI

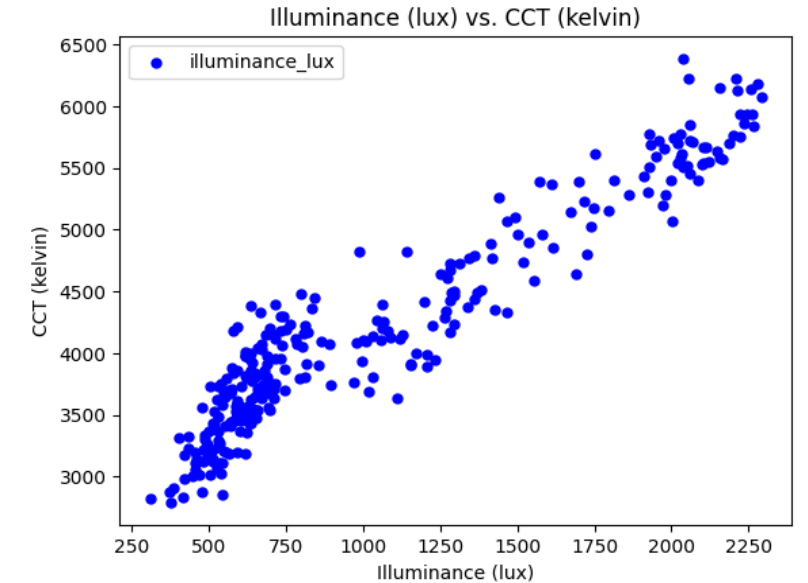
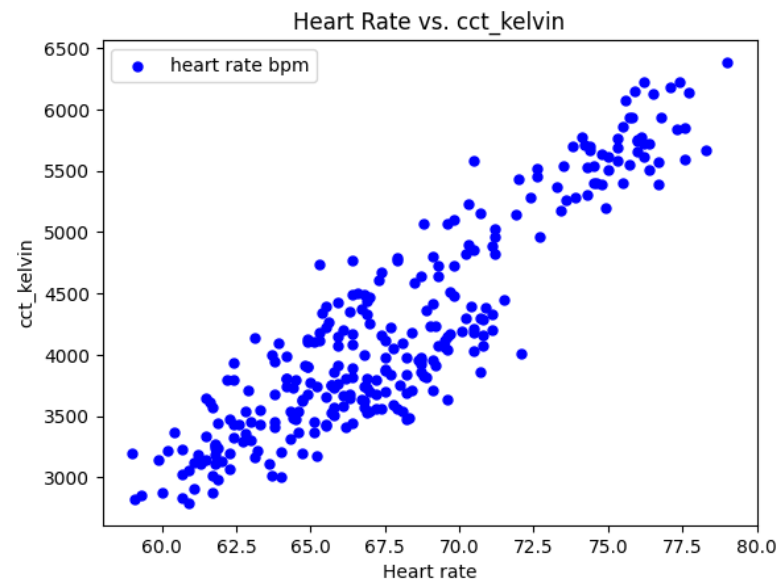
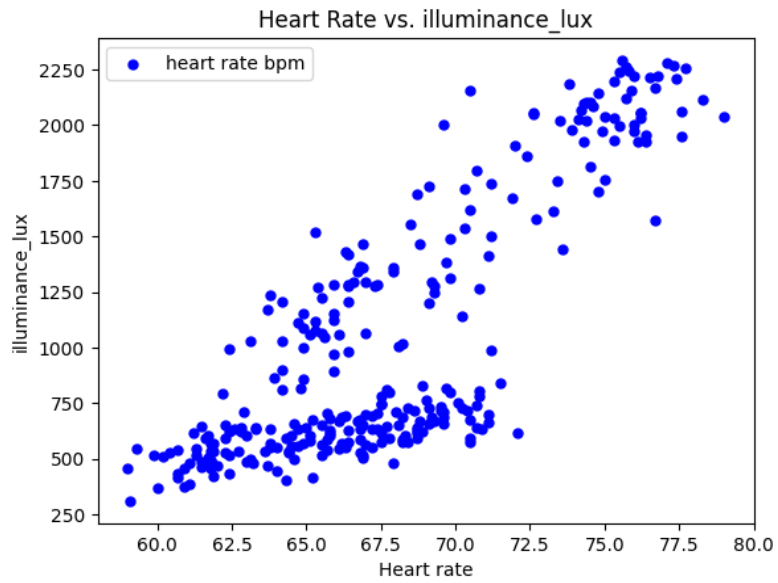


Fig. 15 – DATA output

3 METHODS | AI

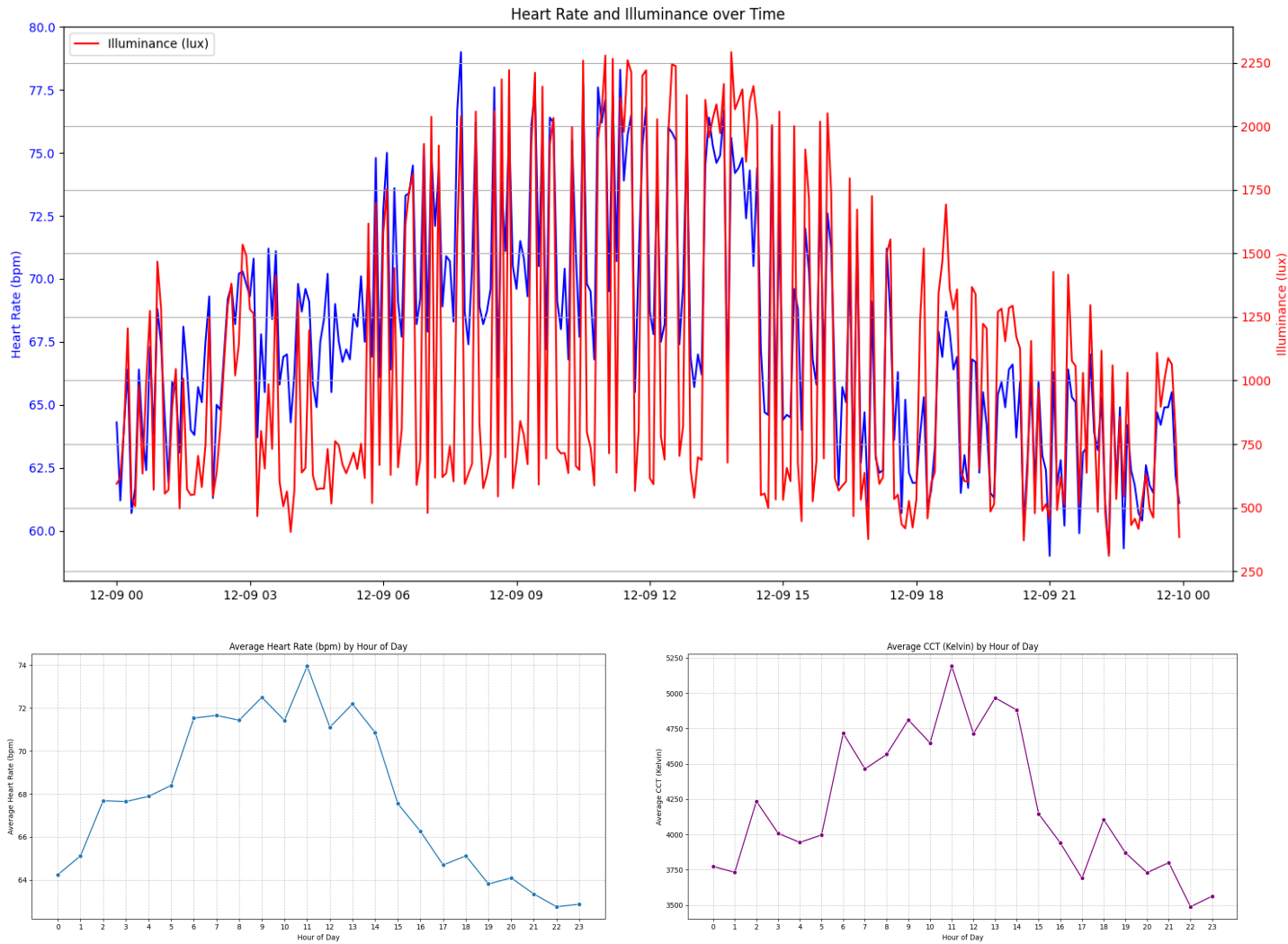


Fig. 16 – DATA output

Fig. 12 – Exterior Vision, AI generated (Nano Banana)



3 METHODS | MATERIAL

Raw materials

- **Bottles** (PET)
- **Food packaging** (ABS, PET, PLA, PP)
- Food waste (more useful for growing plants)
- Kerosine containers (contamination issue)

Fig. 13 – Waste Products



3 METHODS | D2RP

- The project is focused on creating and building with artificial intelligence (AI)-supported
- Design-to-Robotic-Production and Operation (D2RP&O)

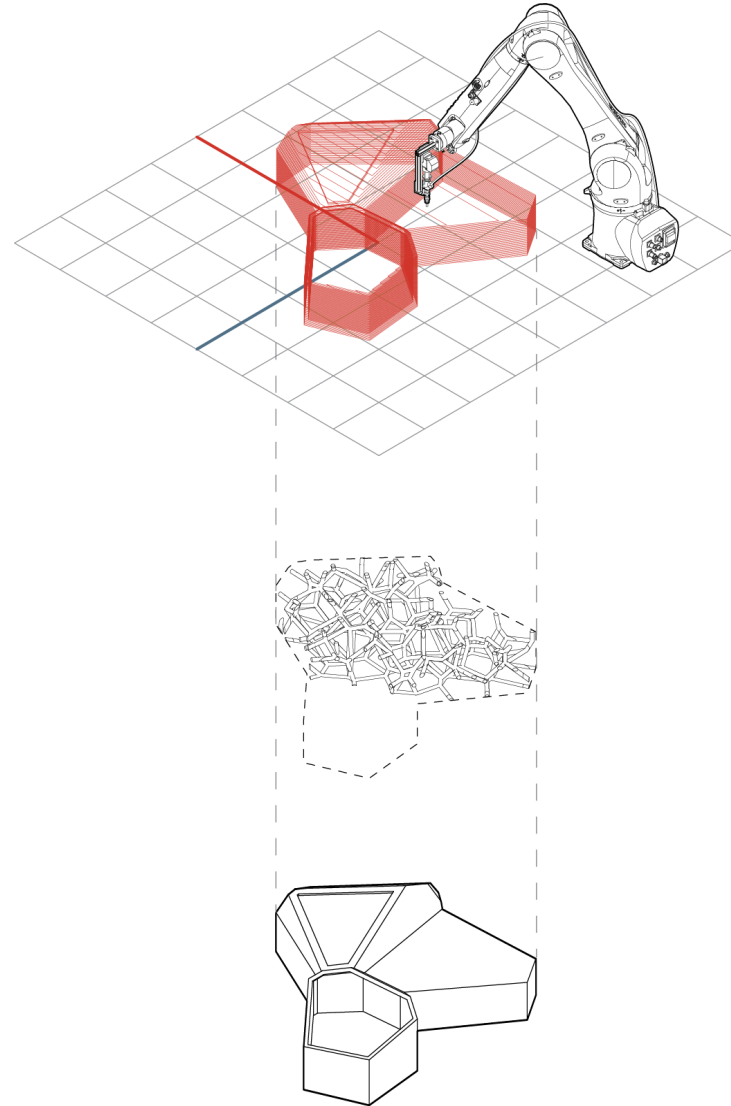


Fig. 15 – Schematic Print Process Fragment

Fig. 14 – KUKA Robot Additive Manufacture

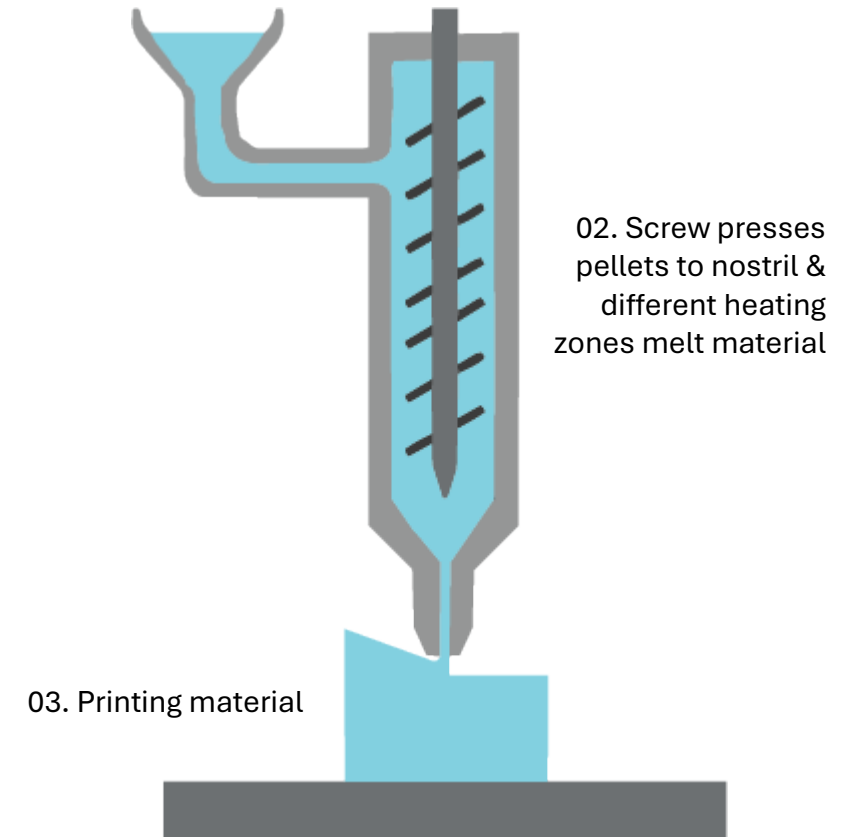


3 METHODS | EXTRUSION TECHNIQUES

Extrusion techniques

- Fused Filament Fabrication (FFF/FDM) -> Filament or pellets
- **Fused particle or pellet fabrication** -> shredded or pelletized waste

01. Pellets are fed into Printer



3 METHODS | ADAPTIVE TECHNOLOGY

Step 1



Bluetooth

The Polar H10 sensor transmits live physiological data using Bluetooth Low Energy (BLE)

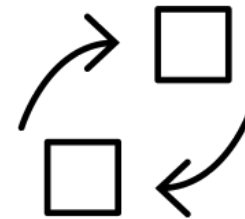
Step 2



Heart rate reception

Python script receives the data, reads the heart rate, maps physiological changes to variations in color temperature (CCT)

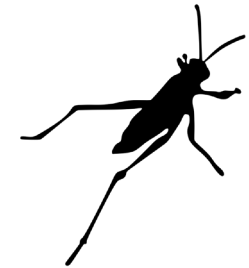
Step 3



CCT to RGB

A data bridge streams the processed RGB data continuously

Step 4



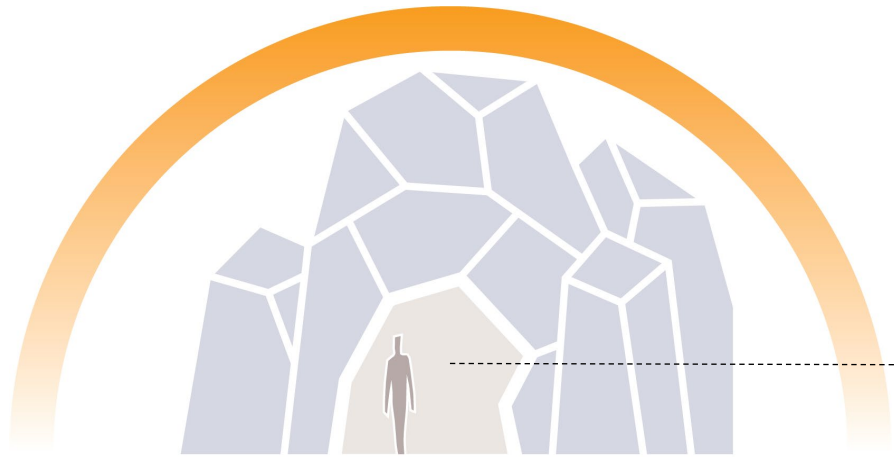
Grasshopper

Grasshopper connects to the RGB output using a client component: GhPython

Fig. 17 – Data implementation steps, source: Python script PolarH10

DESIGN DEVELOPMENT

4 DESIGN DEVELOPMENT | RECAP



CAVE | Protective space

Fig. 19 – Spatial configuration

Group 3

Fig. 18 – Interior perspective



AR122 1:1 Interactive Architecture Prototypes

4 DESIGN DEVELOPMENT | SPATIAL CONFIGURATION

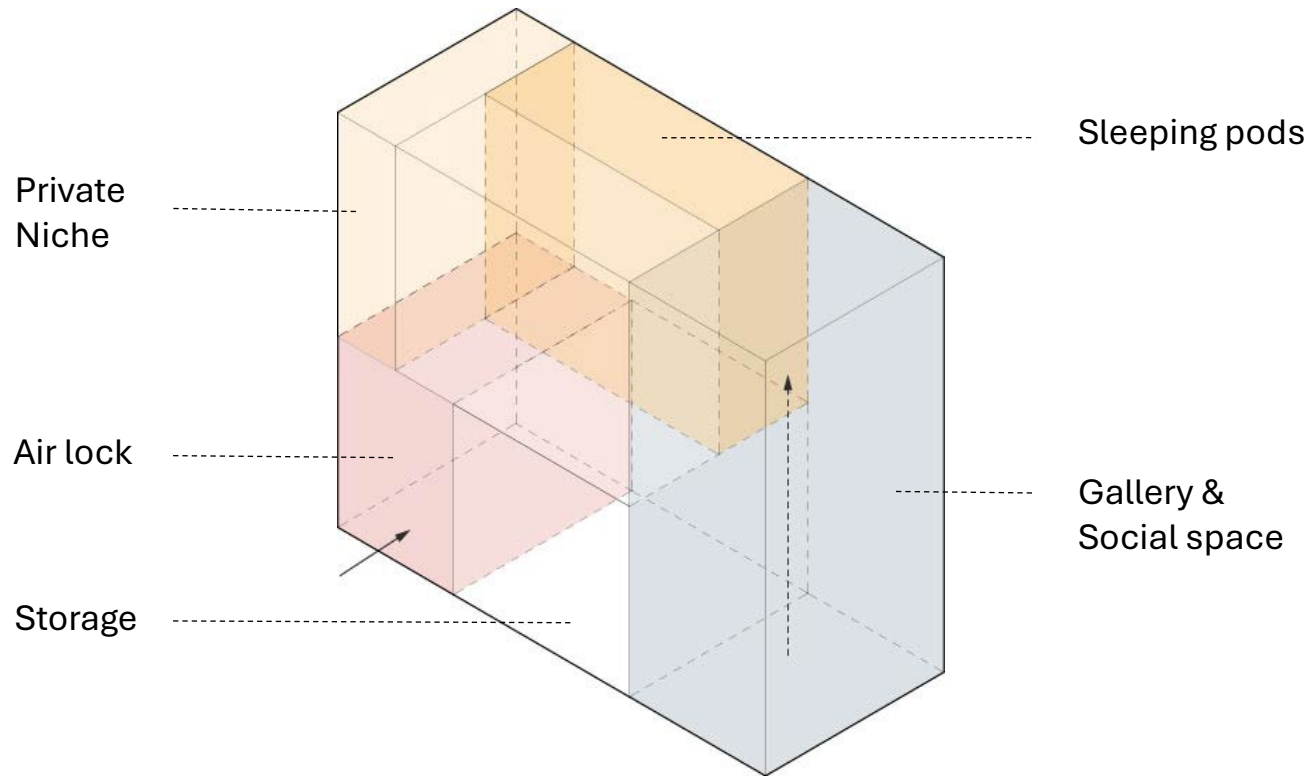
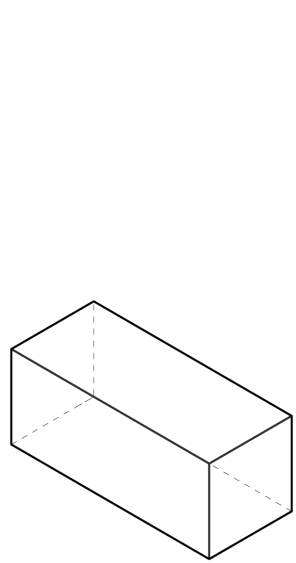


Fig. 20 – Diagram form finding script

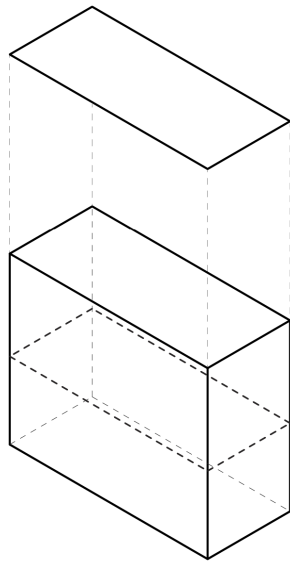
Fig. 18 – Interior perspective



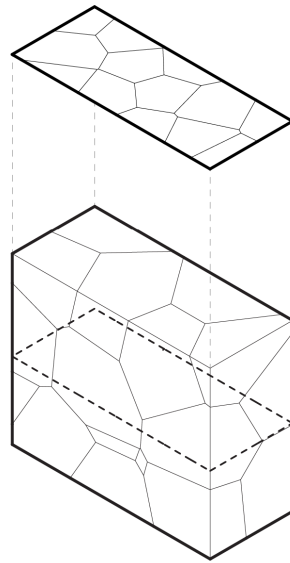
4 DESIGN DEVELOPMENT | SPATIAL CONFIGURATION



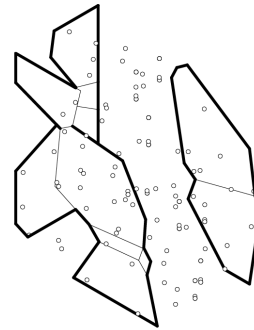
Create container



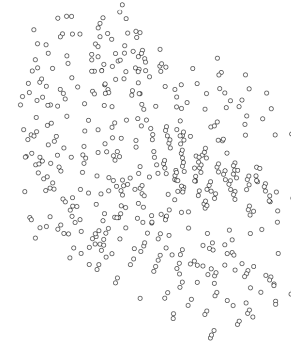
**Stack and
add floor**



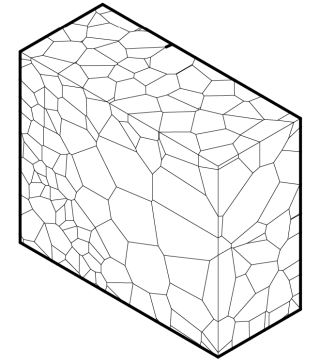
**Apply voronoi
for zoning**



**Pick cells for each density
(S, M, L, XL) and populate**



**Adjust points
manually**



Create spacial voronoi

Fig. 21 – Diagram form finding script

4 DESIGN DEVELOPMENT | GH

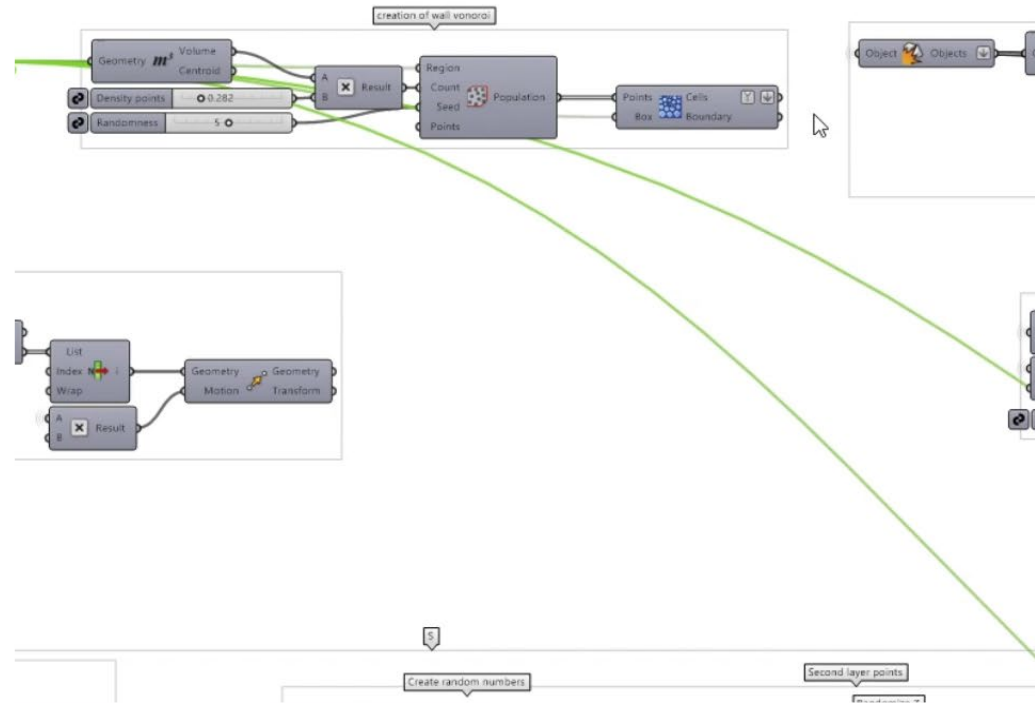
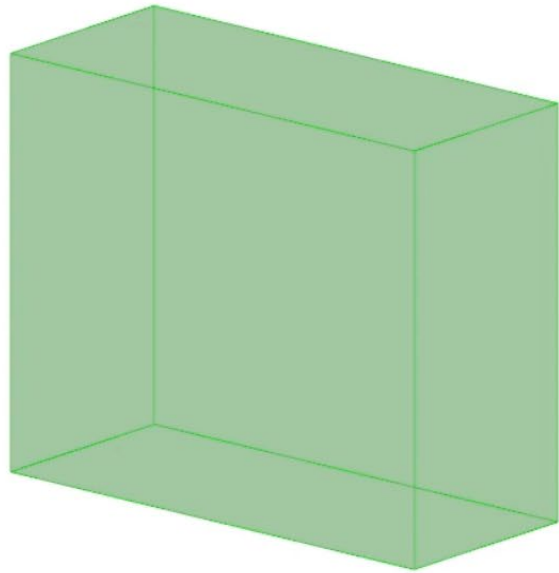


Fig. 22 – Video form finding script

4 DESIGN DEVELOPMENT | SPATIAL CONFIGURATION

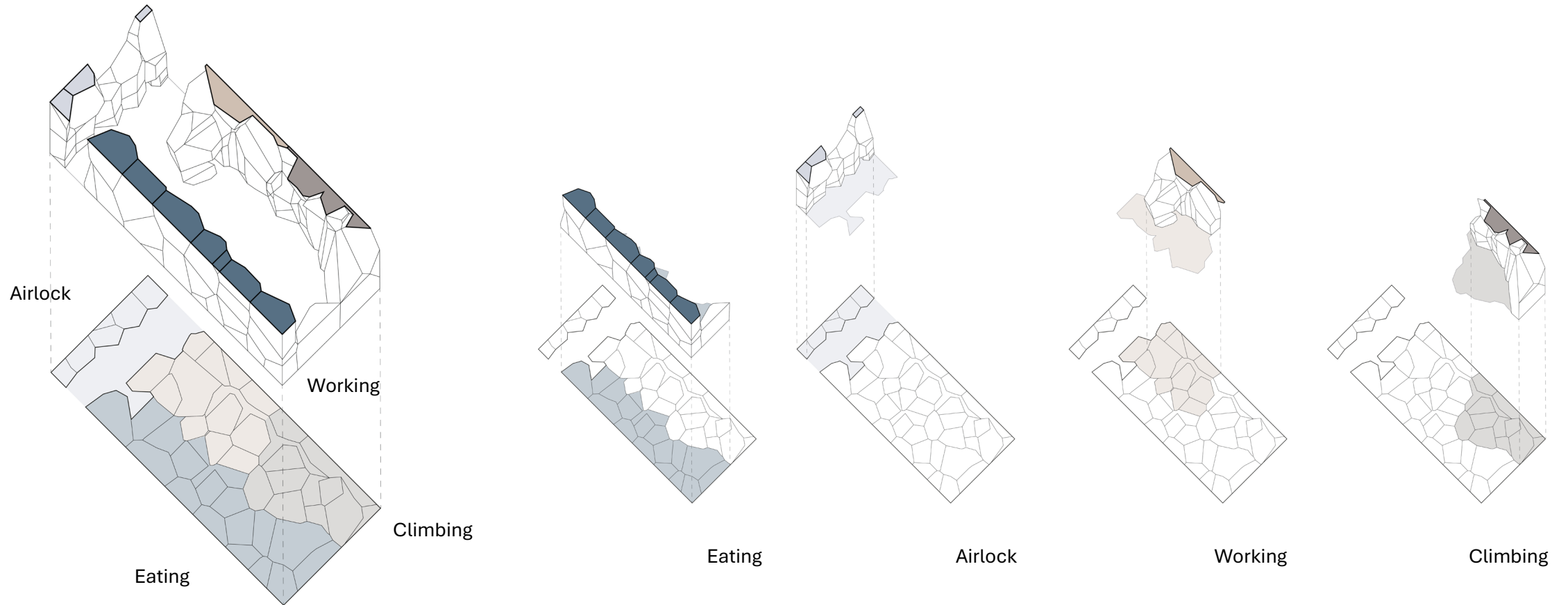


Fig. 23 – Lower Floor Spatial Configuration

4 DESIGN DEVELOPMENT | SPATIAL CONFIGURATION



Fig. 24 – Upper Floor Spatial Configuration

4 DESIGN DEVELOPMENT | FLOOR PLANS

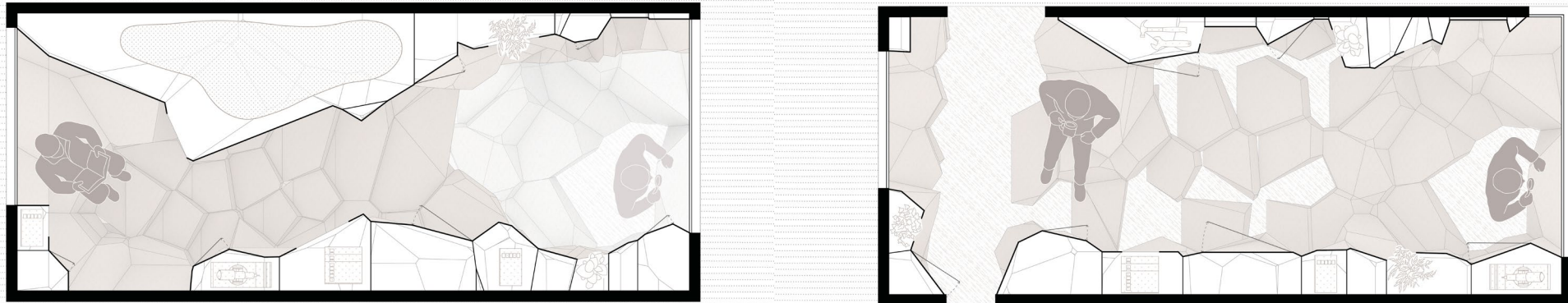


Fig. 25 – Upper and lower floor plans

4 DESIGN DEVELOPMENT | SECTIONS



Fig. 26 – Longitudinal and cross sections

4 DESIGN DEVELOPMENT | SPATIAL CONFIGURATION

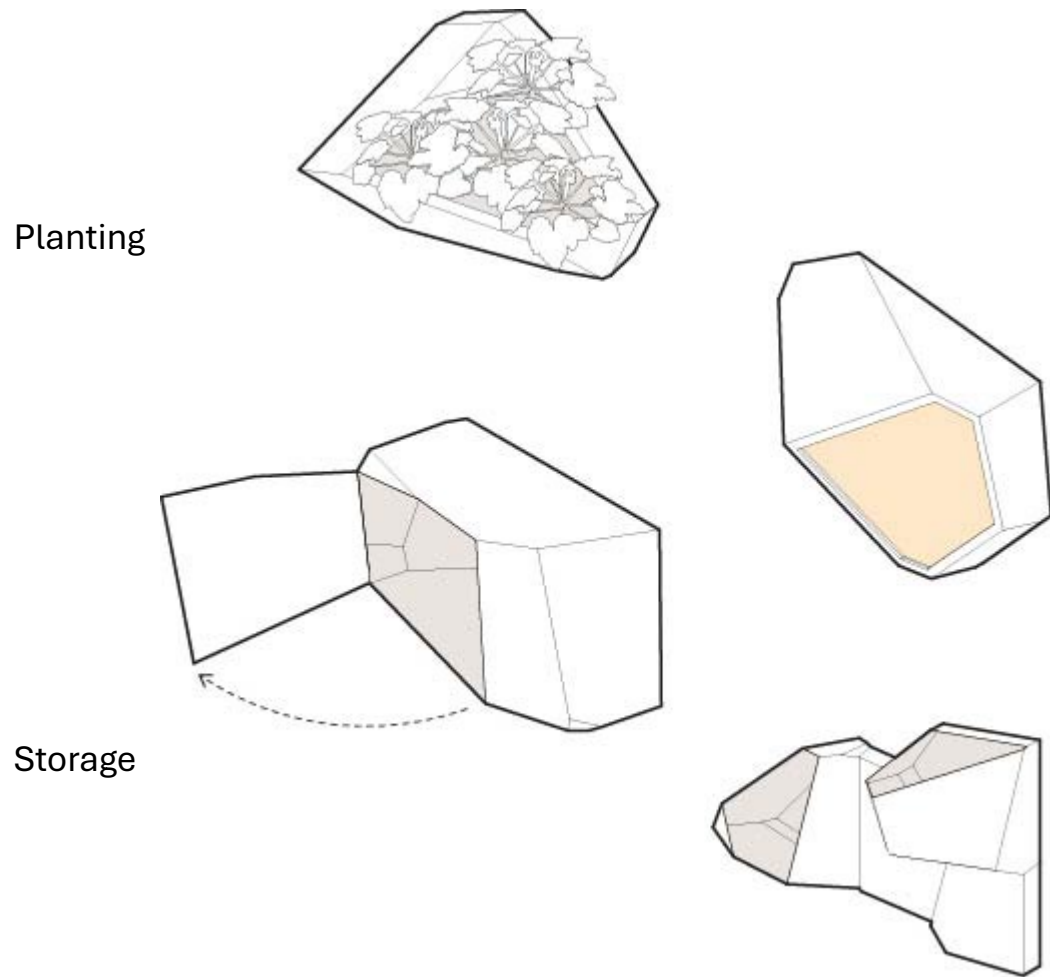


Fig. 27 – Voronoi typologies

Group 3

Lighting

Climbing

Fig. 18 – Interior perspective



AR122 1:1 Interactive Architecture Prototypes

4 DESIGN DEVELOPMENT | **FOLDING CABINETS**

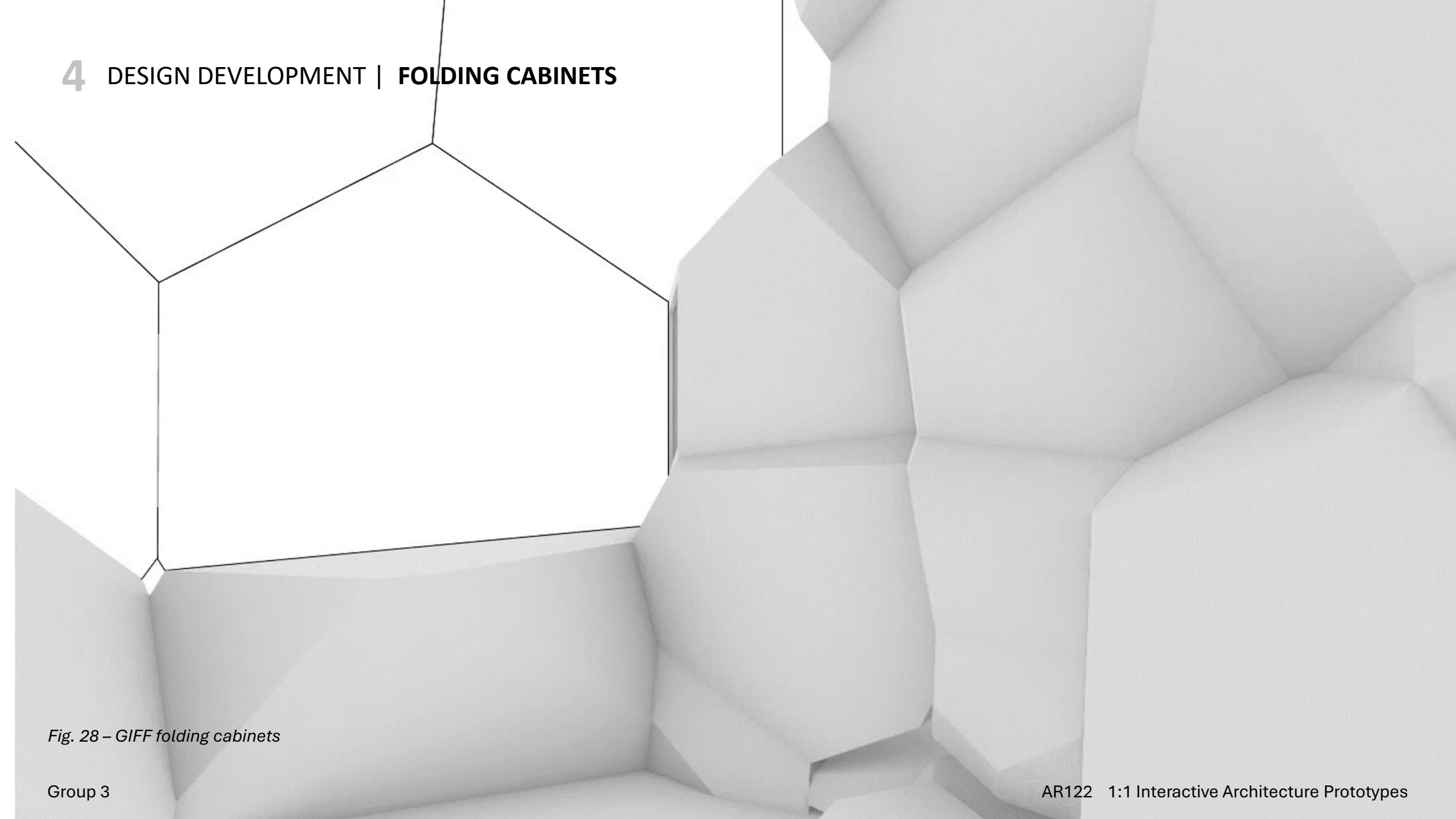


Fig. 28 – GIFF folding cabinets

4 DESIGN DEVELOPMENT | ENTRANCE CABINETS

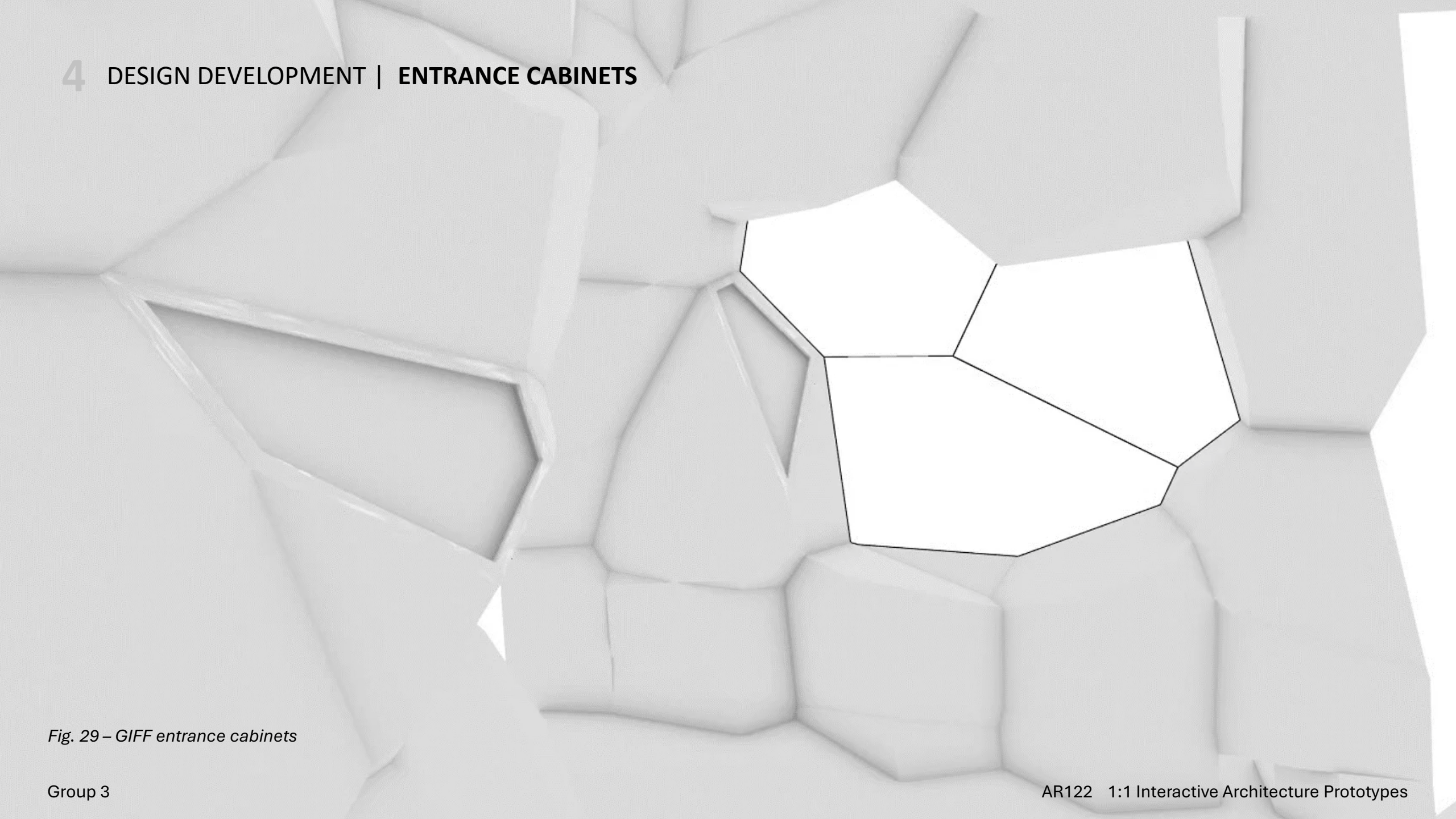


Fig. 29 – GIFF entrance cabinets

4 DESIGN DEVELOPMENT | **PRIVACY NICHE**

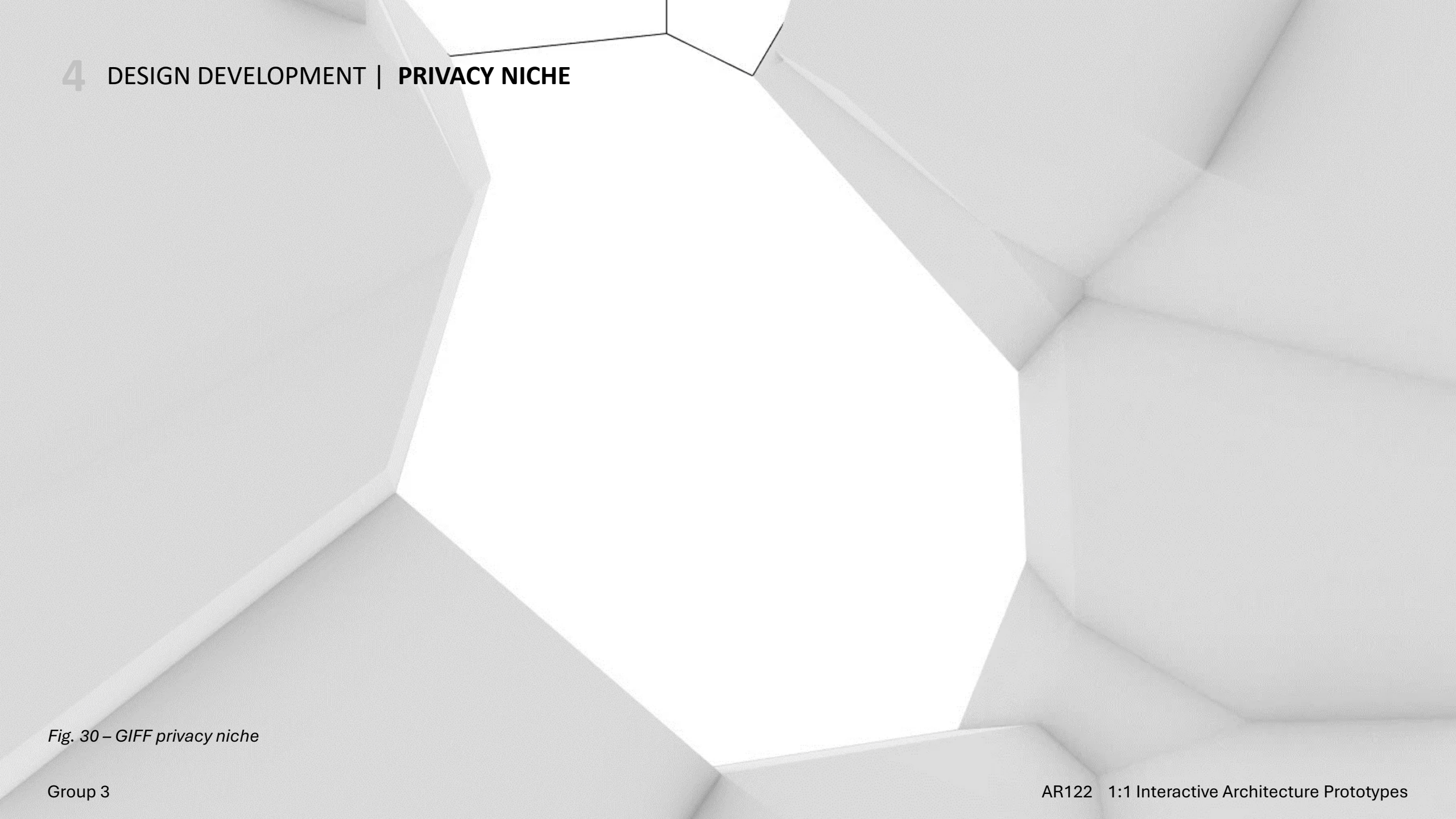


Fig. 30 – GIFF privacy niche

4

DESIGN DEVELOPMENT | **FOLDING TABLE**

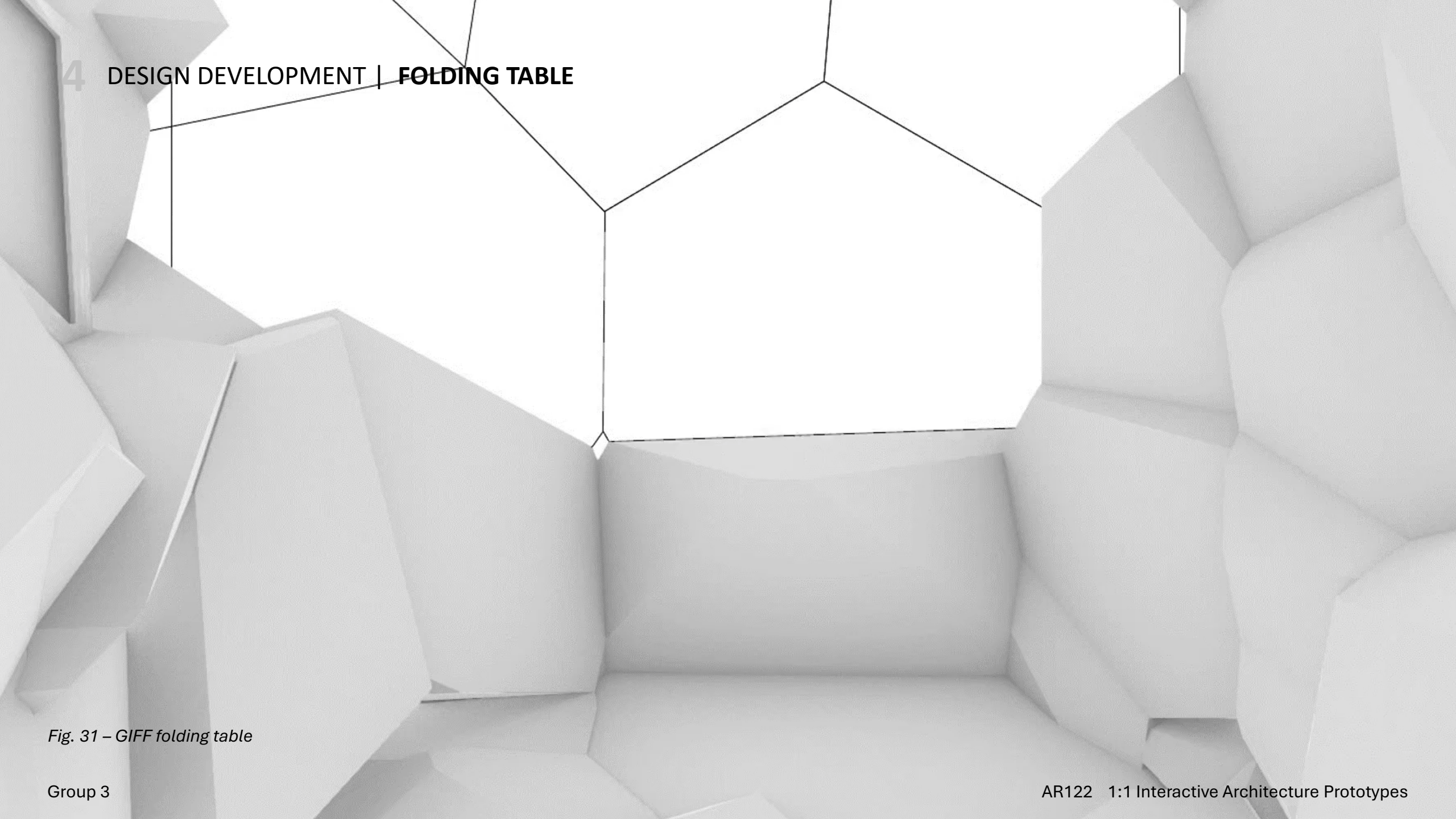


Fig. 31 – GIFF folding table

LIGHTING

5 LIGHTING | CONCEPT

- Simulate circadian rhythm
- Use a high CCT during daytime and working hours
- Gradually shift to low CCT in the morning and evening

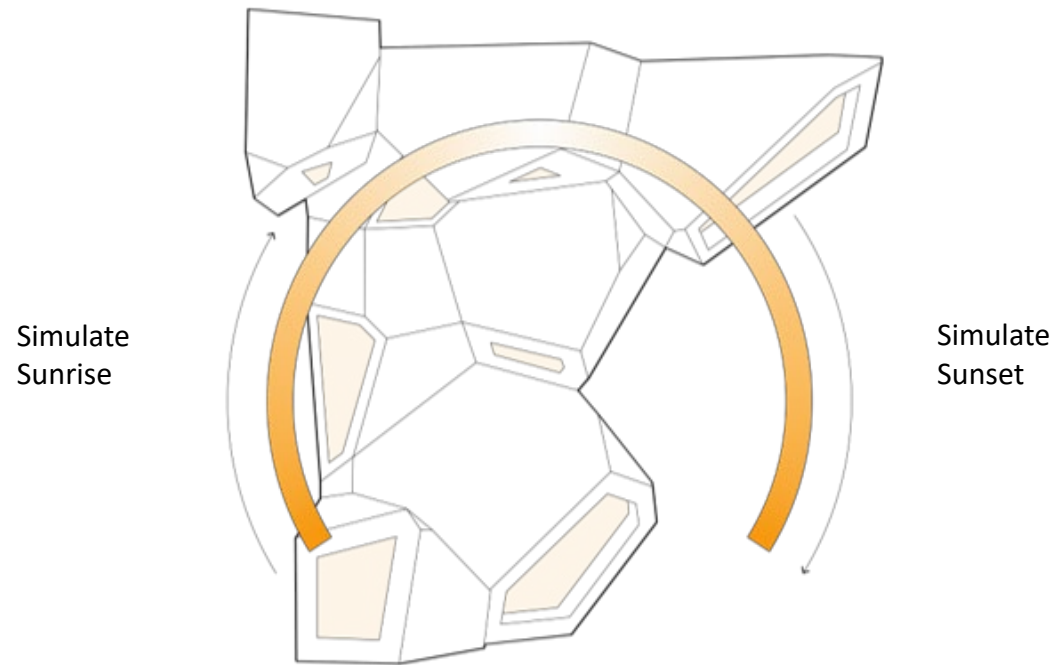


Fig. 32 – Lighting principle

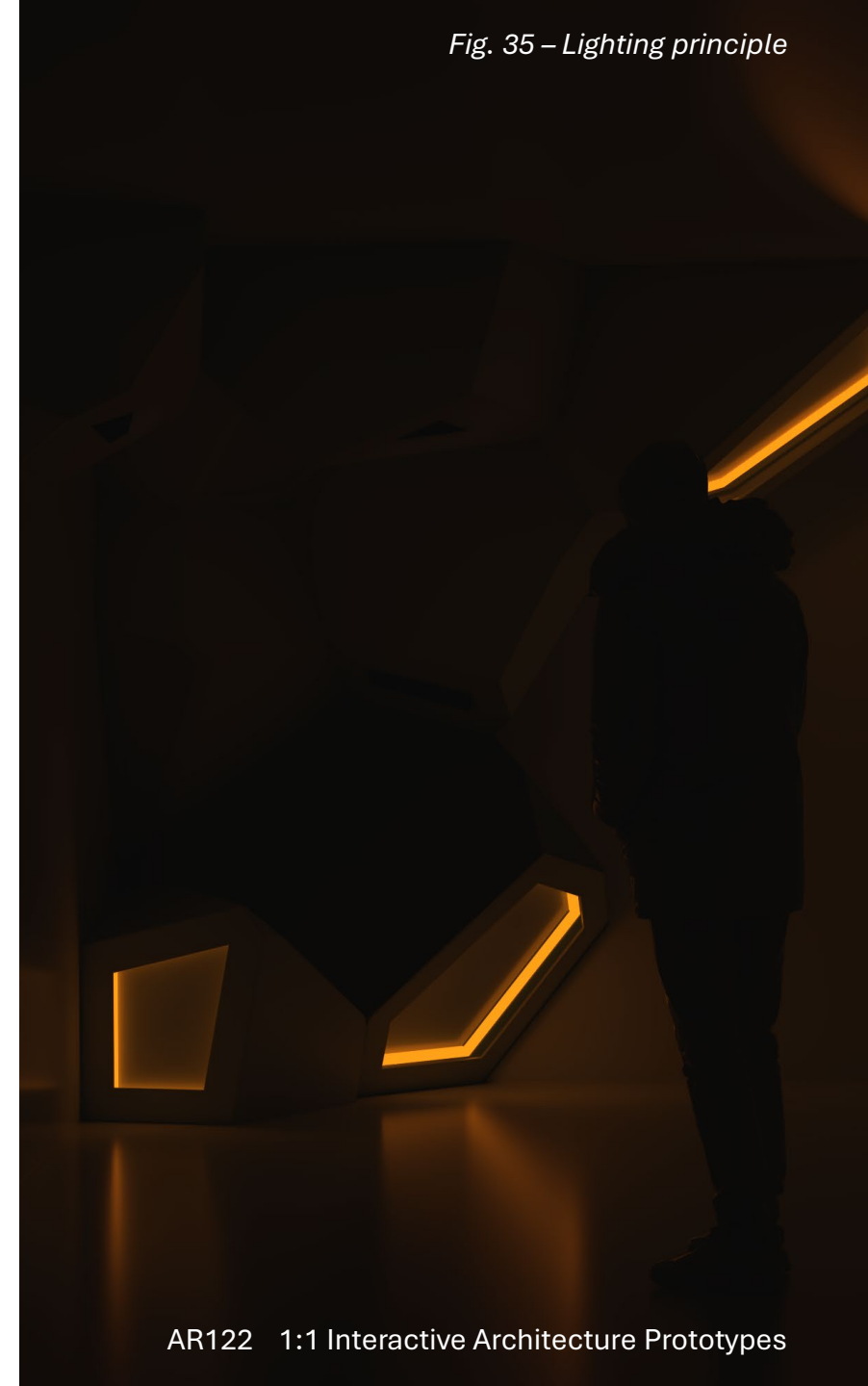


5 LIGHTING | CONCEPT

- Implement heart rate responsive lighting
- When heart rate is lower, the system shifts to a higher CCT
Increased Alertness
- When heart rate is elevated, the system shifts to a lower CCT
Calm Atmosphere



Fig. 34 – Equalizing heartrates



5 LIGHTING | CONCEPT

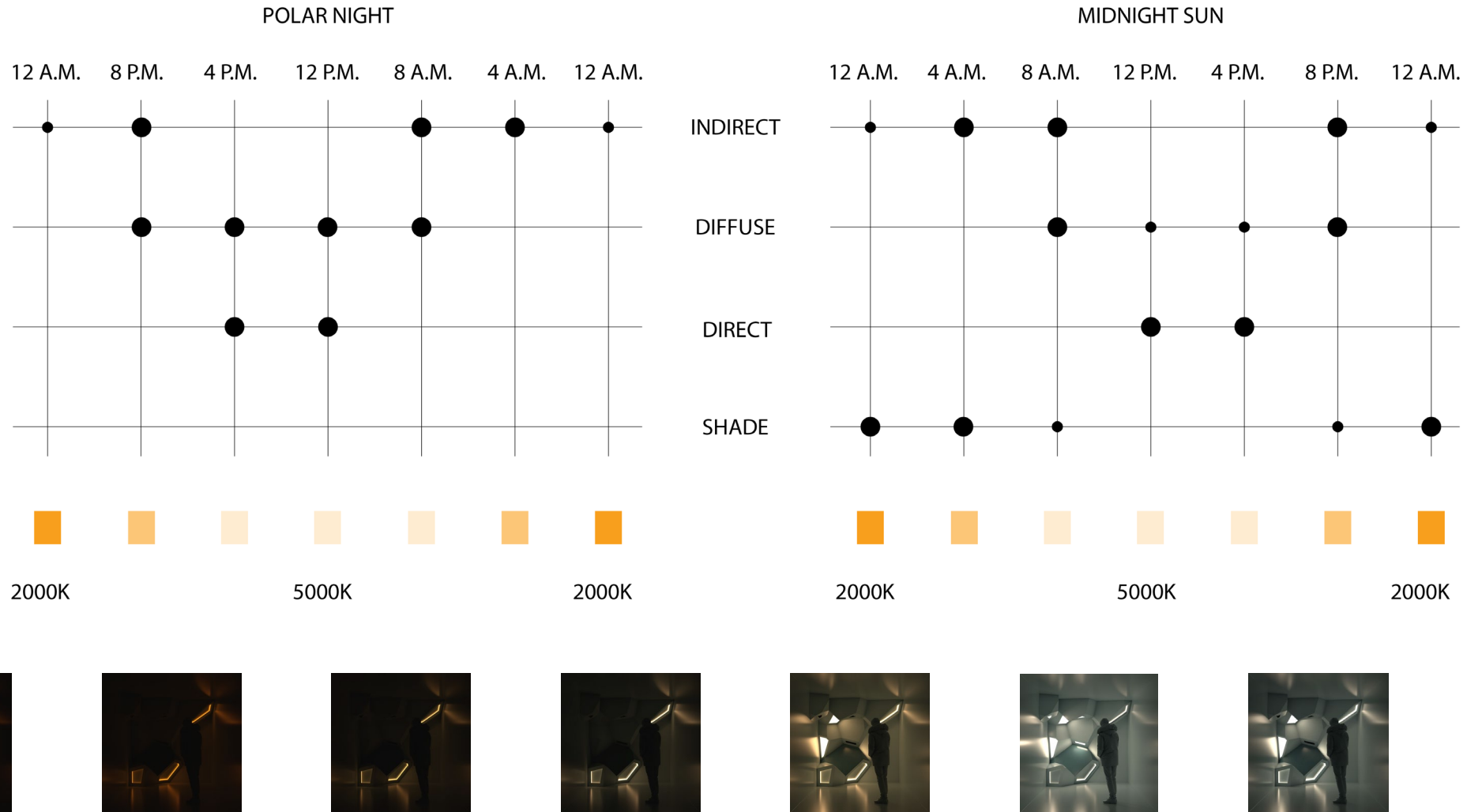


Fig. 36 – CCT adjustment based on activity mapping

5 LIGHTING | GH SCRIPT

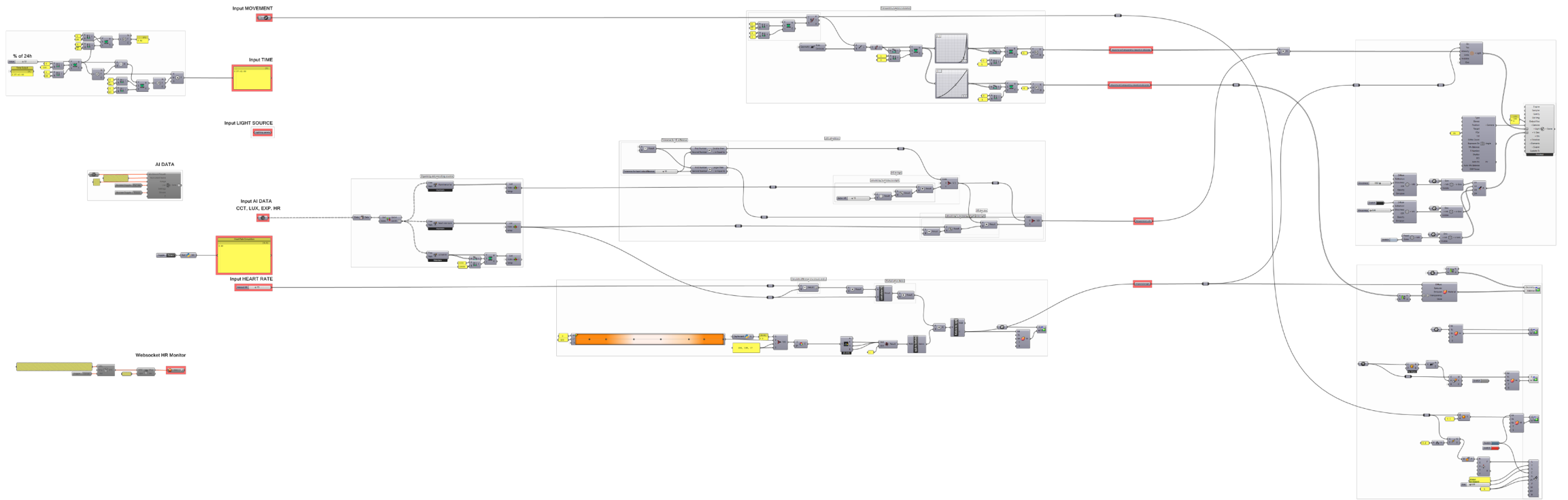


Fig. 37 – Lighting Script Grasshopper - Overview

5 LIGHTING | GH SCRIPT

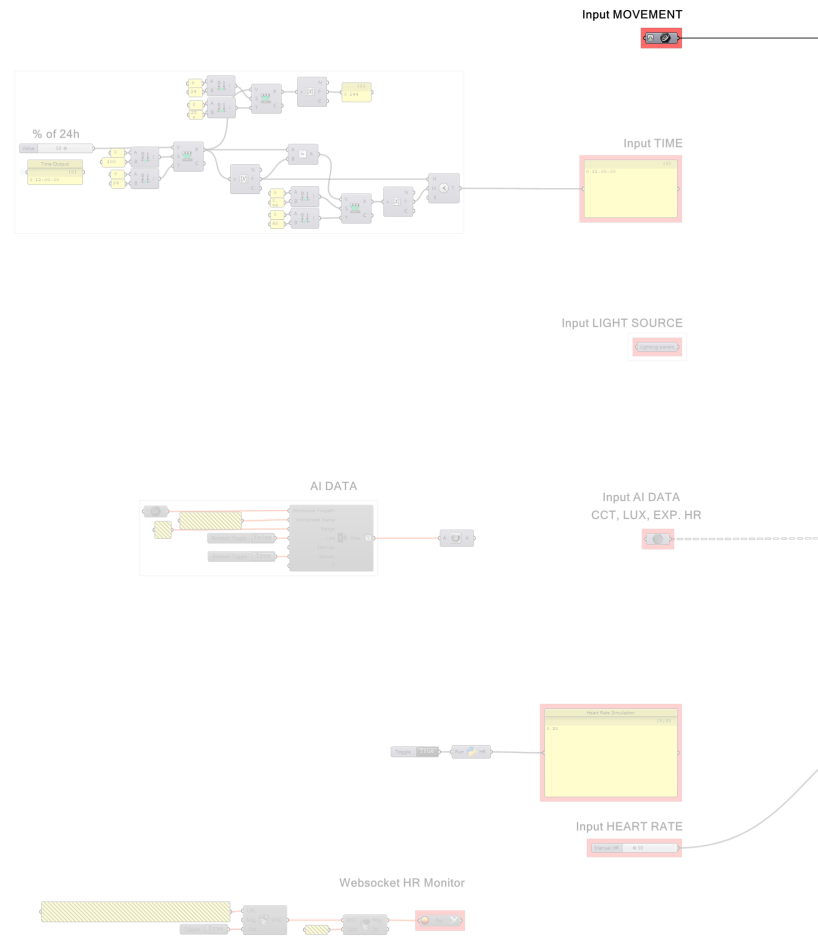
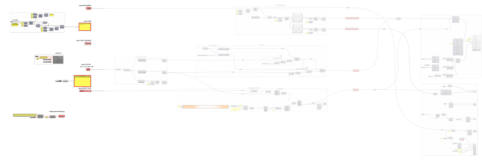


Fig. 38 – Lighting Script Grasshopper – Input Parameters

5 LIGHTING | GH SCRIPT

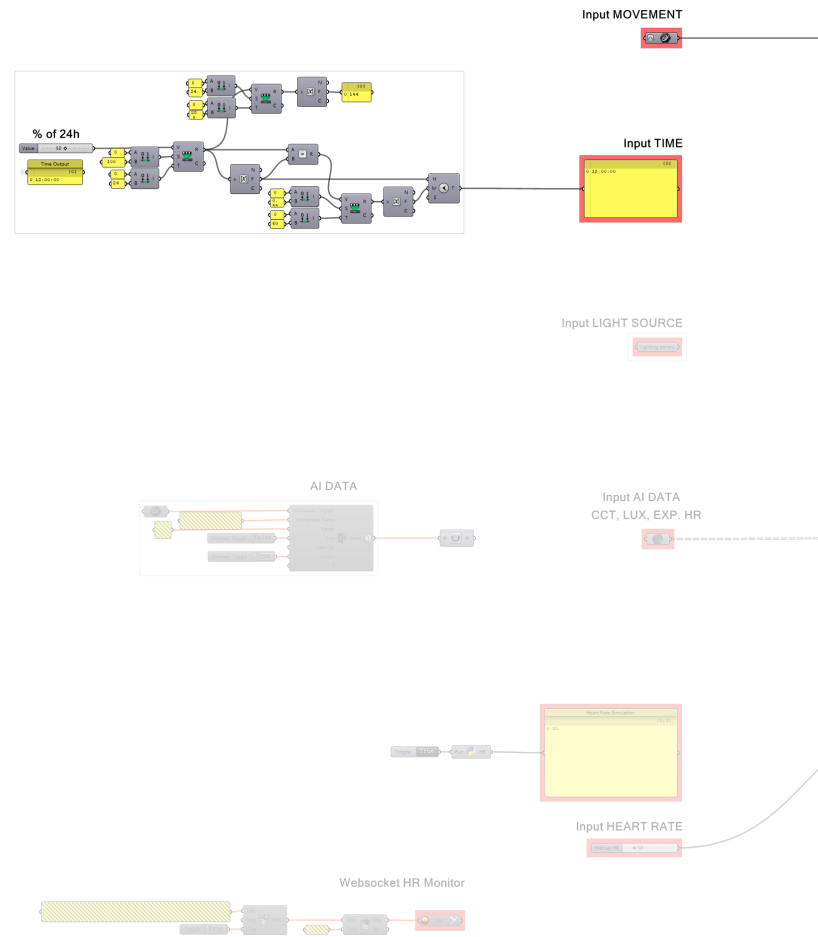
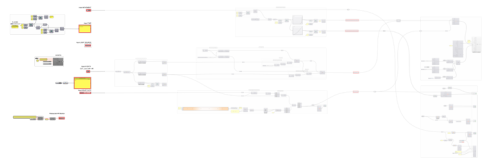


Fig. 38 – Lighting Script Grasshopper – Input Parameters

5 LIGHTING | GH SCRIPT

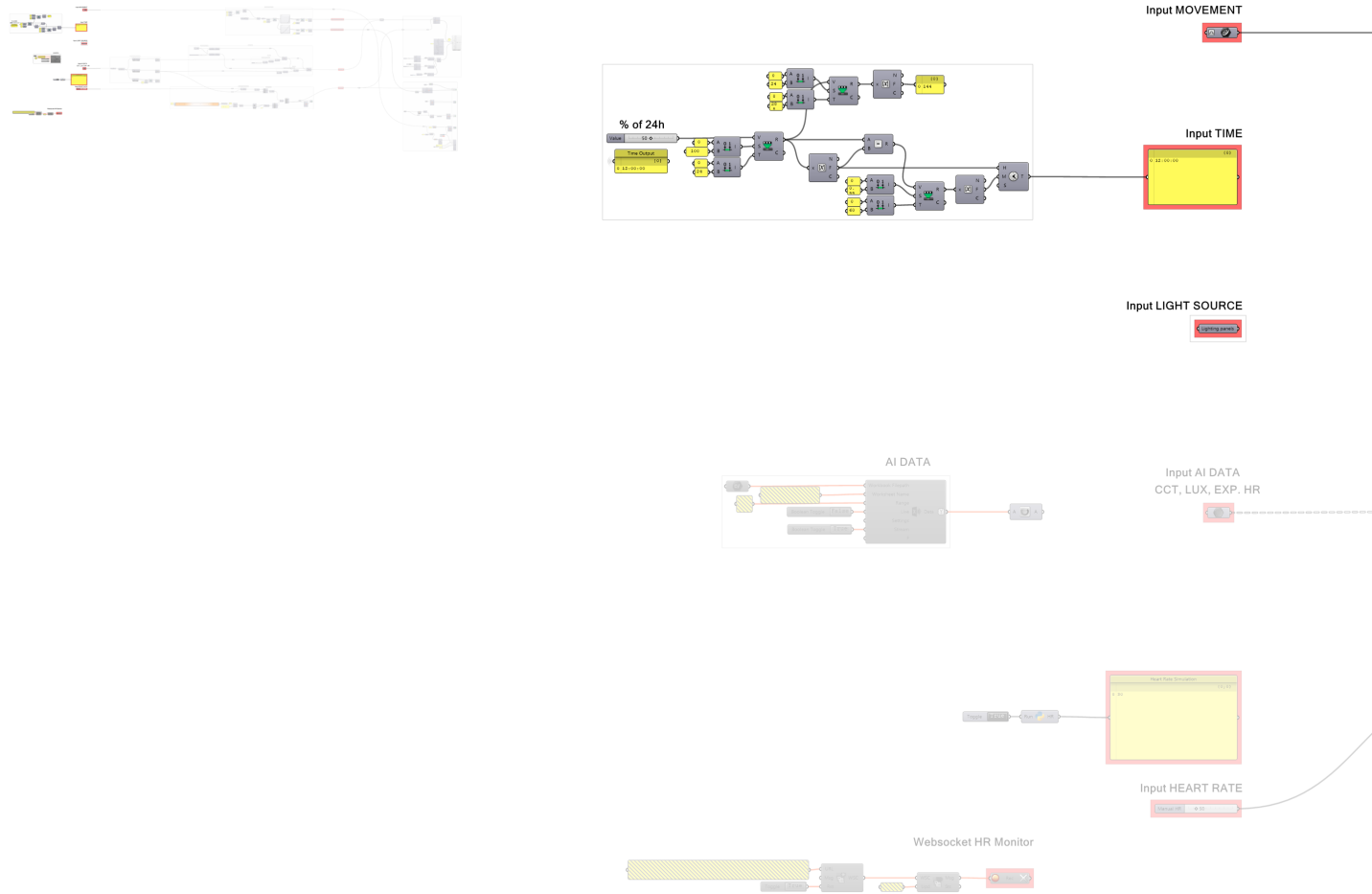


Fig. 38 – Lighting Script Grasshopper – Input Parameters

5 LIGHTING | GH SCRIPT

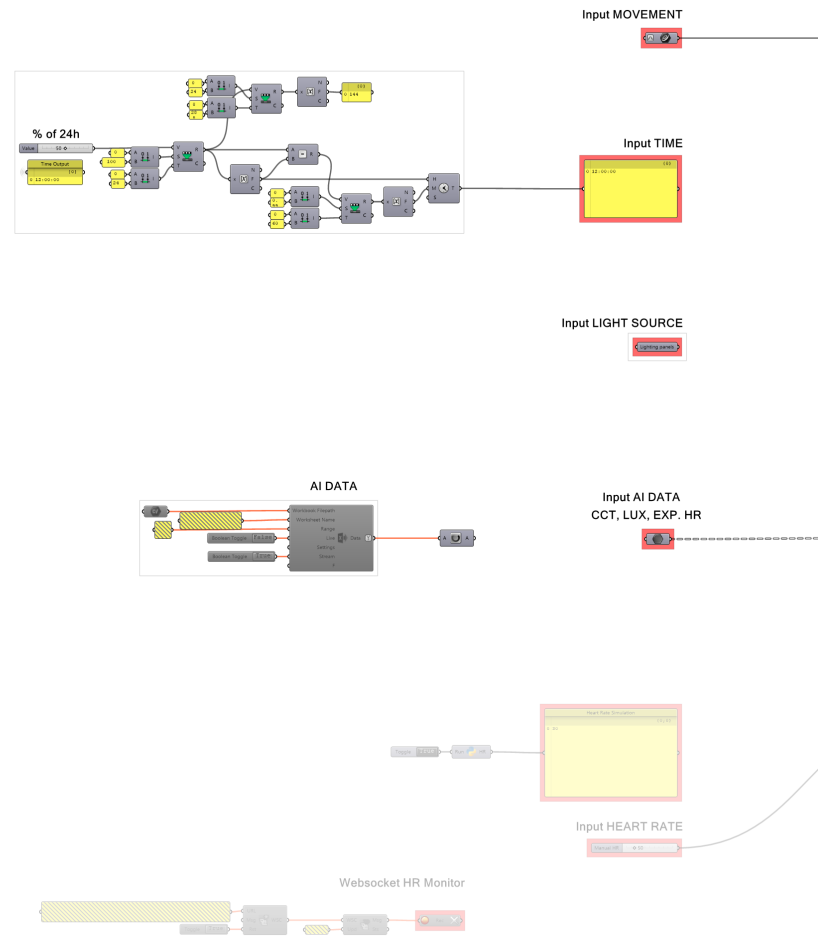
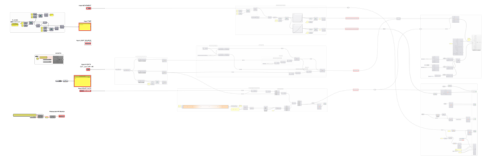


Fig. 38 – Lighting Script Grasshopper – Input Parameters

5 LIGHTING | GH SCRIPT

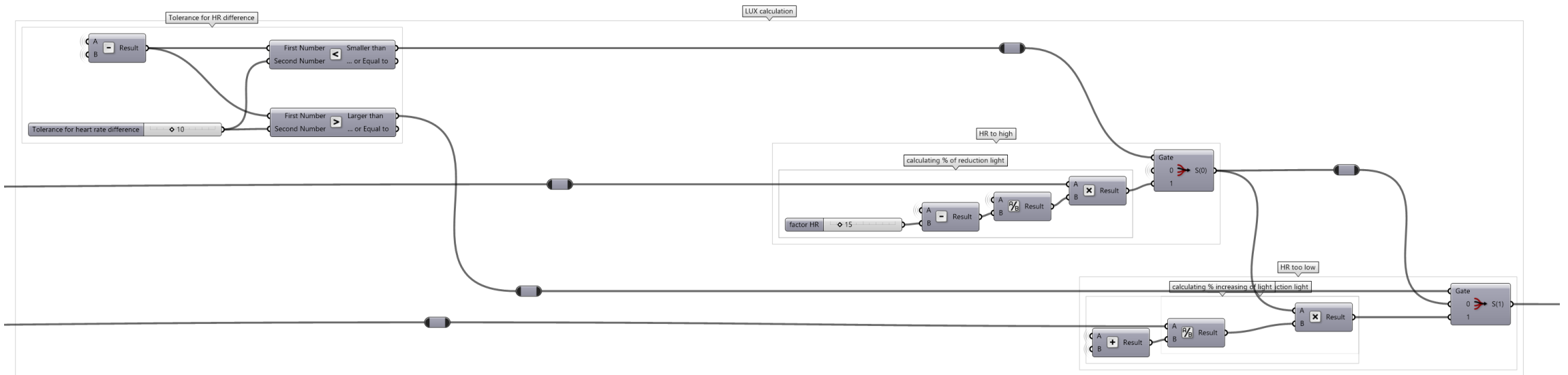
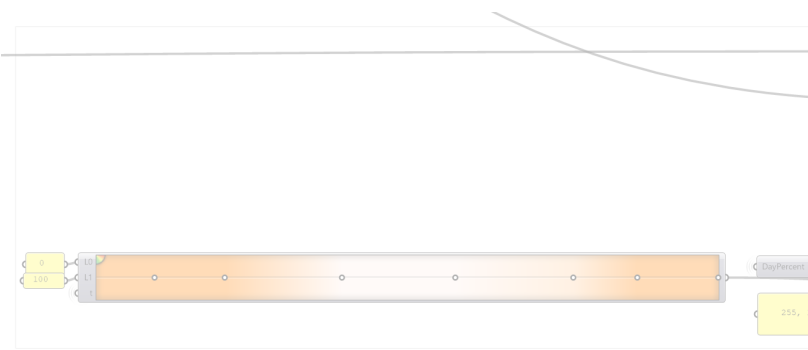
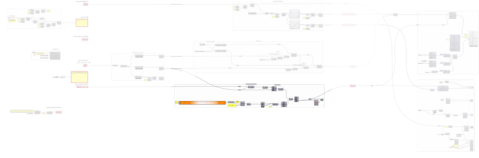


Fig. 41 – Lighting Script Grasshopper – Calculation tolerance threshold

5 LIGHTING | GH SCRIPT



```

1 import math
2
3 def kelvin_to_rgb(K):
4     K = max(1000.0, min(40000.0, float(K)))
5     T = K / 100.0
6
7     if T <= 66:
8         R = 255
9     else:
10        R = 329.698727446 * ((T - 60) ** -0.1332047592)
11
12    if T <= 66:
13        G = 99.4708025861 * math.log(T) - 161.1195681661
14    else:
15        G = 288.1221695283 * ((T - 60) ** -0.0755148492)
16
17    if T >= 66:
18        B = 255
19    elif T <= 19:
20        B = 0
21    else:
22        B = 138.5177312231 * math.log(T - 10) - 305.0447927307
23
24    R = max(0, min(255, R))
25    G = max(0, min(255, G))
26    B = max(0, min(255, B))
27
28    return (R, G, B)
29
30 def color_error(rgb1, rgb2):
31     return ((rgb1[0] - rgb2[0]) ** 2 +
32            (rgb1[1] - rgb2[1]) ** 2 +
33            (rgb1[2] - rgb2[2]) ** 2)
34
35 def parse_rgb(rgb_input):
36     if isinstance(rgb_input, str):
37         parts = rgb_input.split(",")
38         if len(parts) != 3:
39             raise ValueError("RGB string must look like '255,255,255'")
40         return tuple(float(p.strip()) for p in parts)
41
42     if hasattr(rgb_input, "_iter_"):
43         vals = list(rgb_input)
44         if len(vals) != 3:
45             raise ValueError("RGB input must have 3 values")
46         return (float(vals[0]), float(vals[1]), float(vals[2]))
47     raise ValueError("unsupported RGB input format")
48
49 target_rgb = parse_rgb(068)
50 target_rgb = (
51     max(0.0, min(255.0, target_rgb[0])),
52     max(0.0, min(255.0, target_rgb[1])),
53     max(0.0, min(255.0, target_rgb[2]))
54 )
55
56 best_K = 6500.0
57 best_err = float("inf")
58
59 for K in range(1000, 40001, 100):
60     err = color_error(target_rgb, kelvin_to_rgb(K))
61     if err < best_err:
62         best_err = err
63         best_K = float(K)
64
65 start_K = max(1000.0, best_K - 200.0)
66 end_K = min(40000.0, best_K + 200.0)
67
68 K = start_K
69 while K <= end_K:
70     err = color_error(target_rgb, kelvin_to_rgb(K))
71     if err < best_err:
72         best_err = err
73         best_K = K
74     K += 1.0
75
76 kelvin = float(best_K)
77
78 print(kelvin)

```

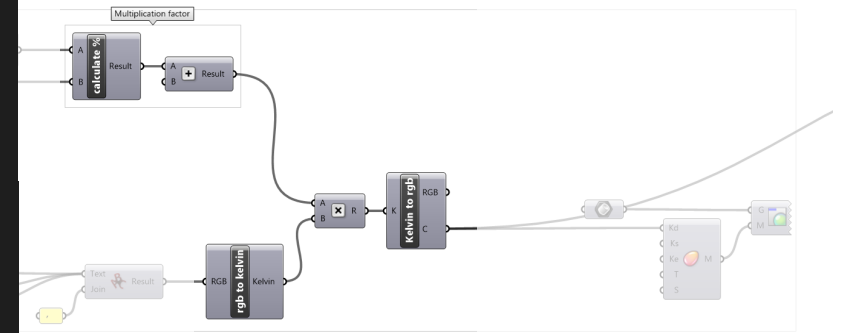


Fig. 43 – Tamal-sen/colourTemperatureToRGB (GitHib)

Fig. 42 – Lighting Script Grasshopper - Heart Rate–Driven colour Temperature Adjustment

5 LIGHTING | GH SCRIPT

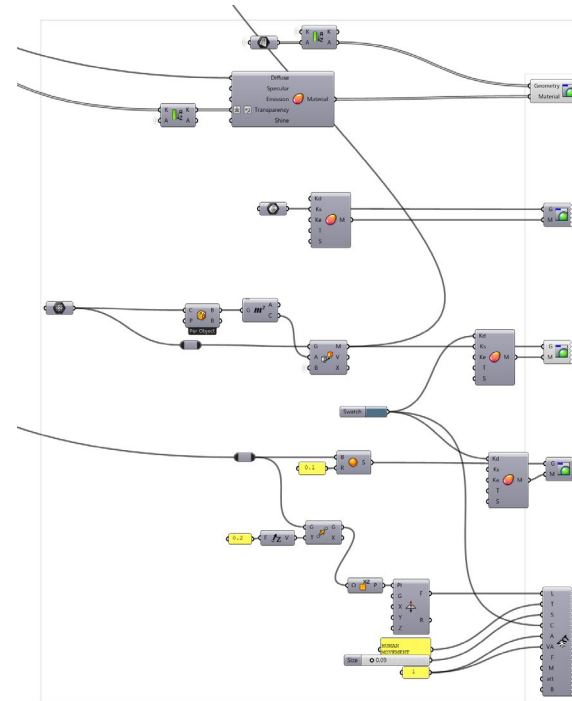
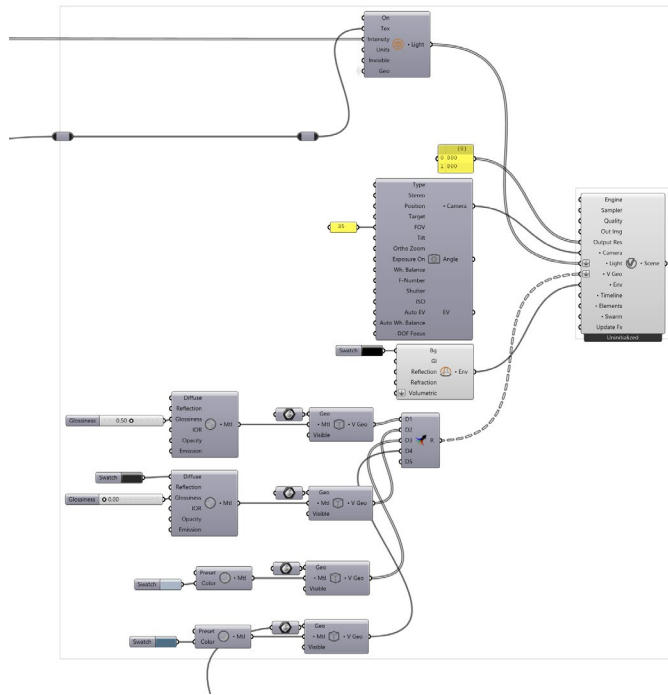


Fig. 44 – Overview Lighting Script Grasshopper - Visualisation using Chaos V-Ray & Rhino Preview

5 LIGHTING | GH SCRIPT

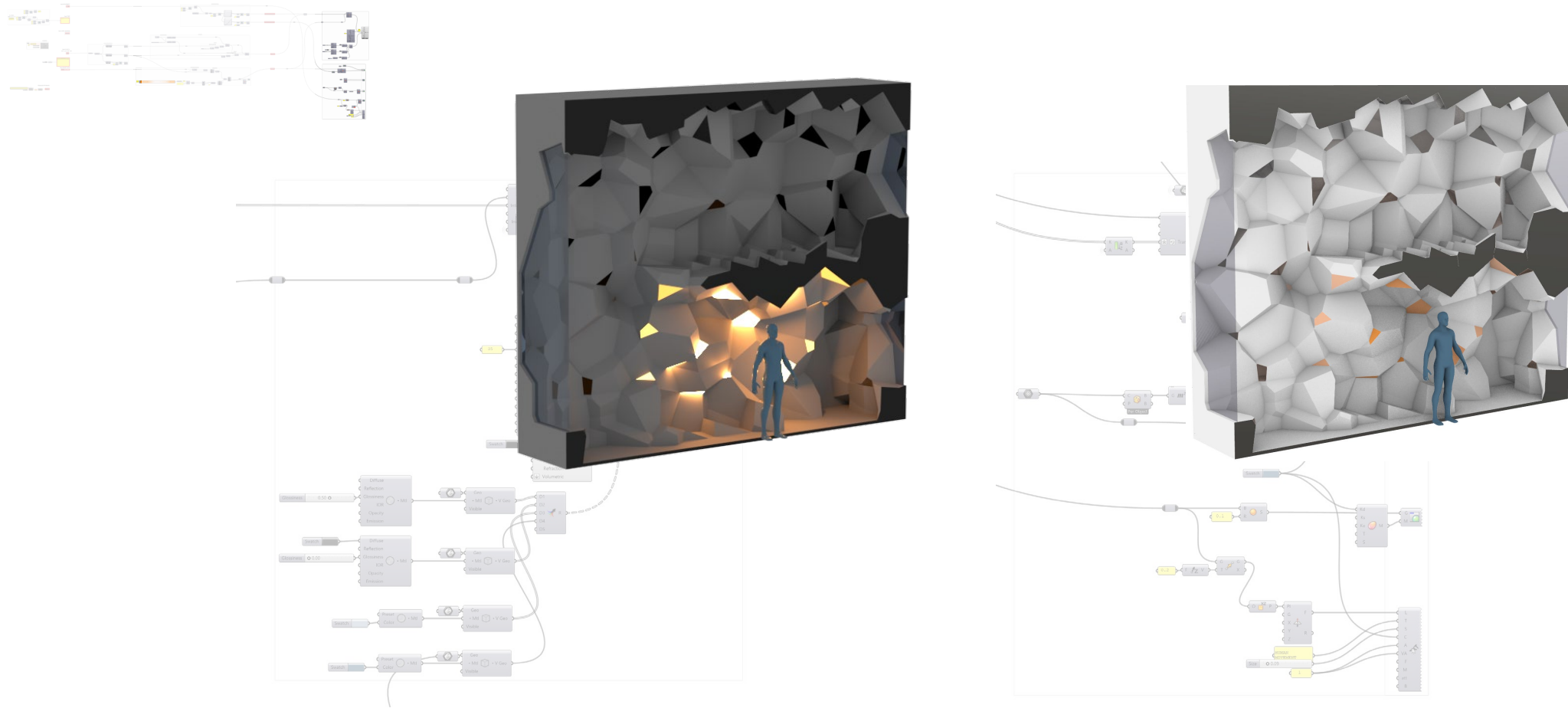


Fig. 45 – Chaos V-Ray & Rhino Preview

5 LIGHTING | DEMONSTRATION

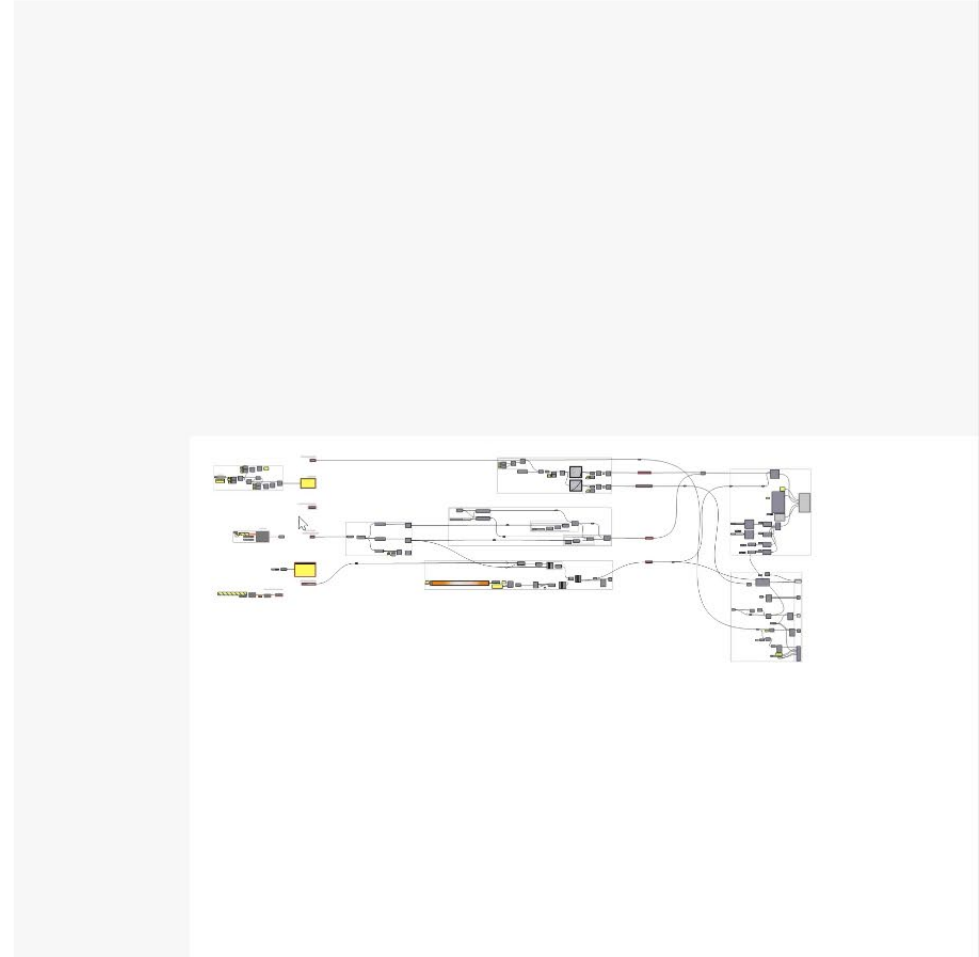
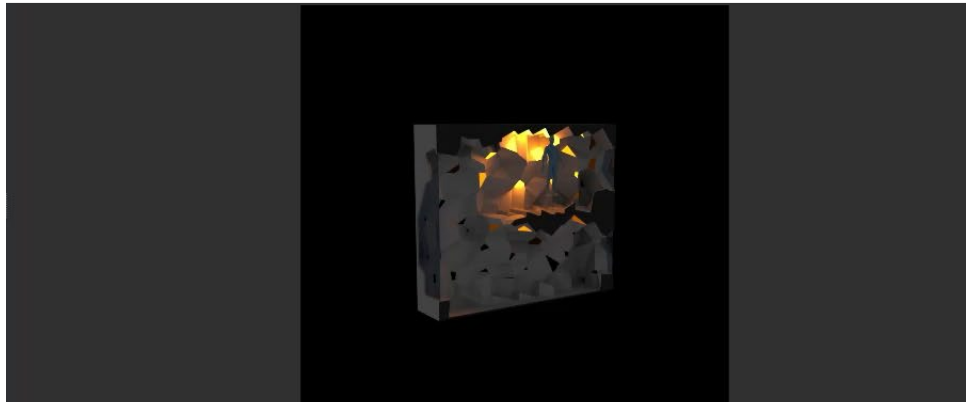
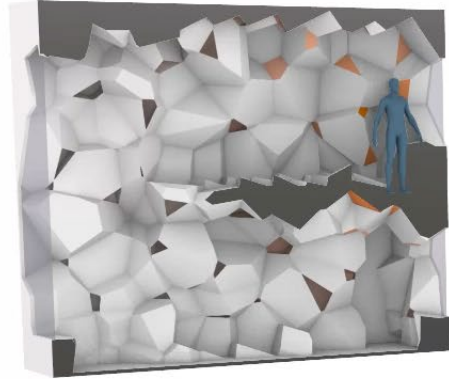


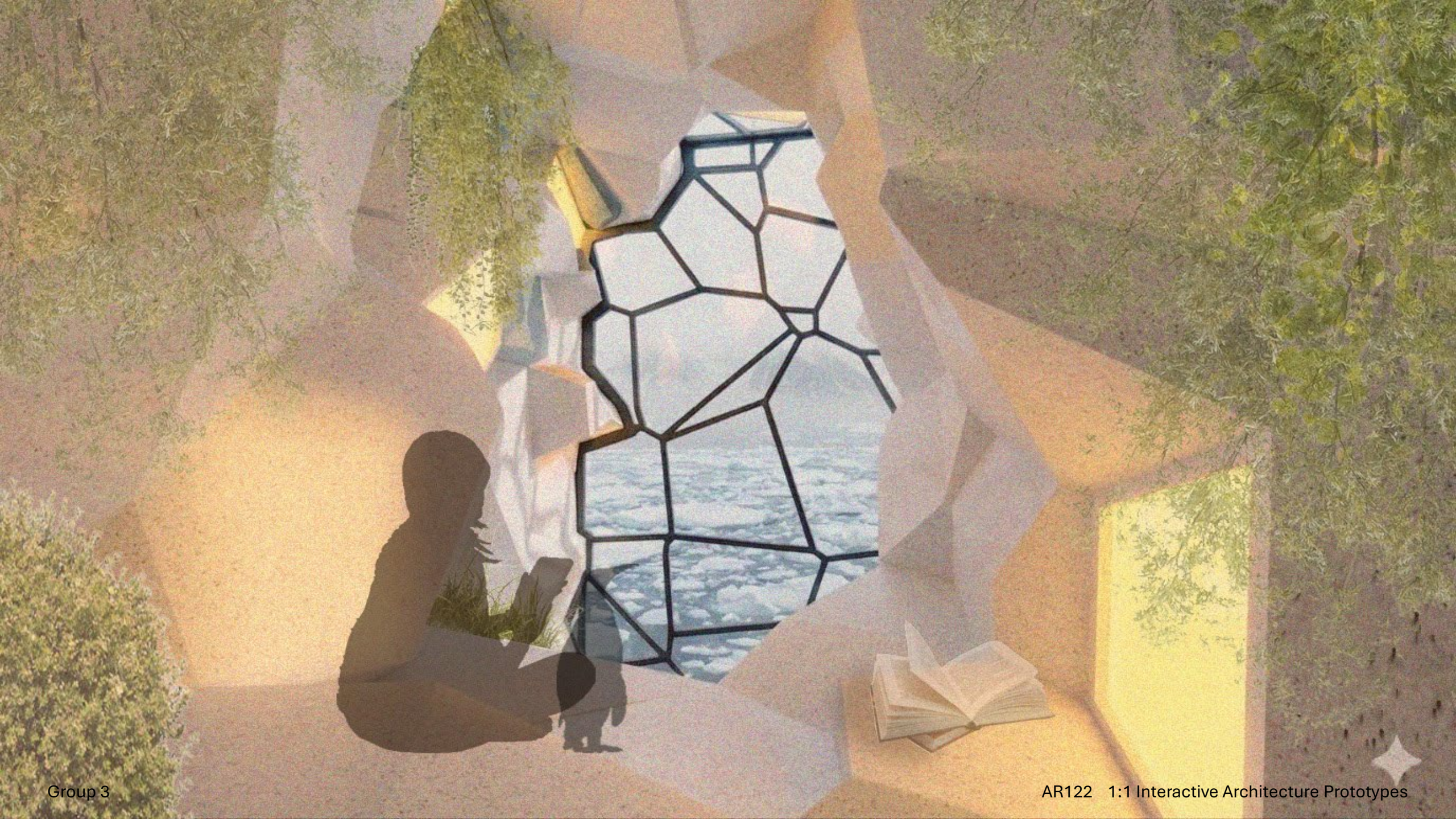
Fig. 46 – Lighting Script Grasshopper – Lighting script demonstration

CPLX

Cave Pulse Light Experiment

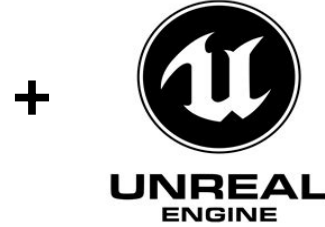
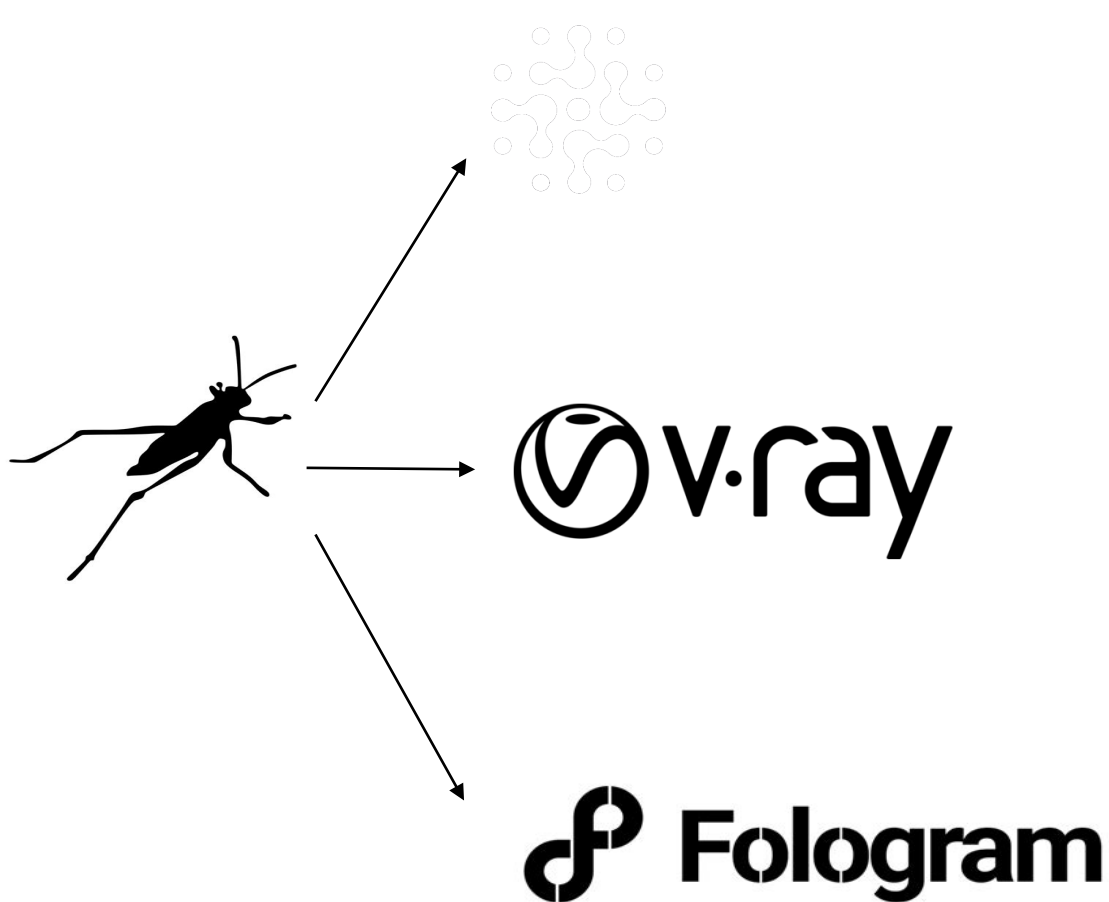






LIMITATION

7 LIMITATIONS | LIGHT VISUALIZATION



ISSUES

No option for light preview, only material display

No live connection for lighting

No option for light preview, only material display

Fig. XX – Limitations in light visualization tools

CPLX

Cave Pulse Light Experiment

THE GOAL



THE NECESSITY



THE STRATEGY



THE OUTPUT

IS

To design



a **human-centered**
interior

for the

TROLL Norwegian
Research Station in
Antarctica

using

6m x 2.5m shipping
container modules

IS

To improve HEALTH by



- **Mitigating Chronic
Stress**

caused by
Extreme Isolation

- **Aligning Circadian
Disruption**

caused by
Lack of natural day-night
cycles

- **Prioritizing Experience-
First Design**

IS

To use



- **Voronoi-based geometry**

as a
Reconfigurable Design Logic

- **AI-assisted lighting**

with an
Adaptive Pulse system

- **Circular Fabrication**

to
Convert plastic waste into 3D-
printing filament

IS

To provide



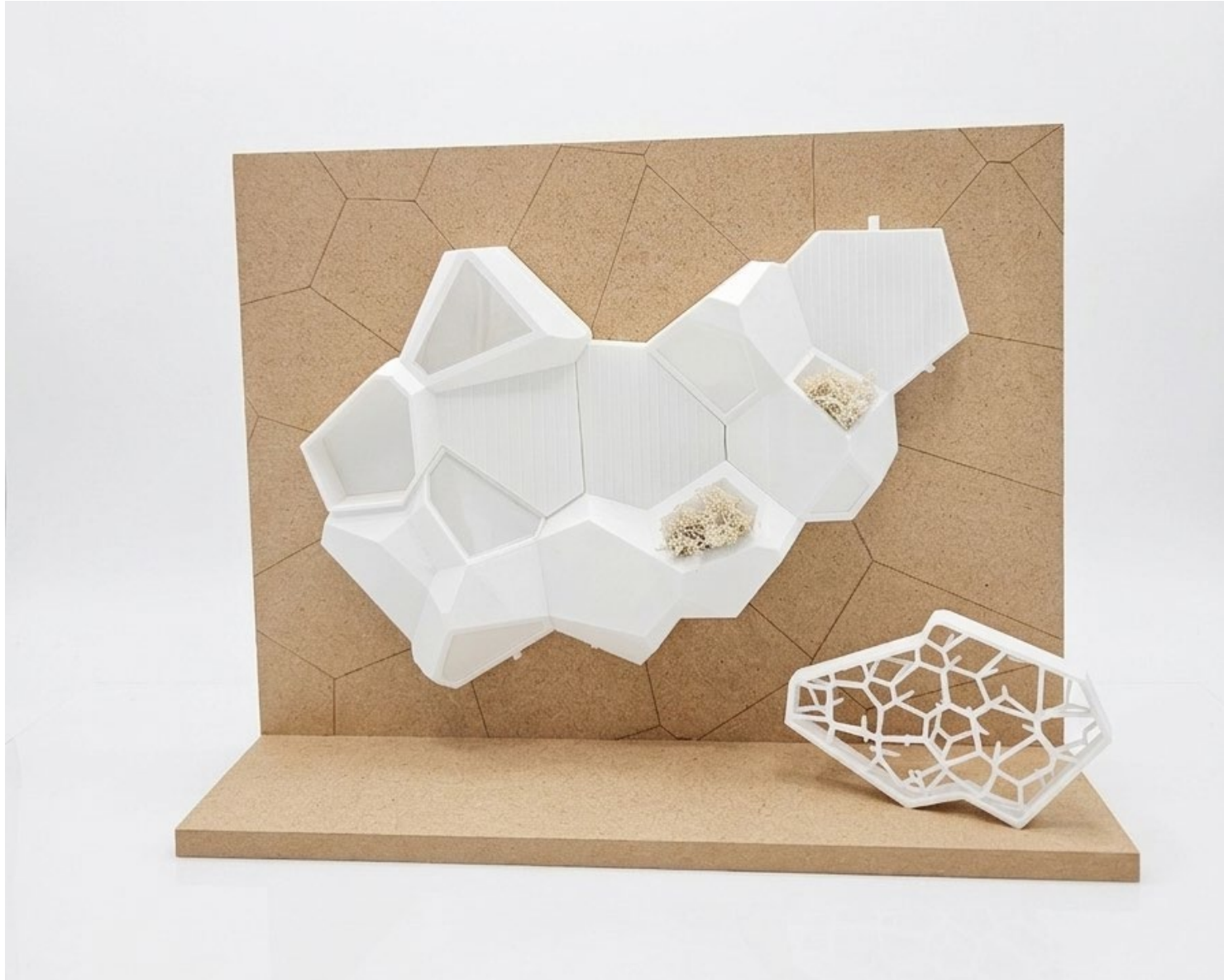
- **Cave interior** supporting
sleeping, working, socializing
with
Biophilic elements and
integrated climbing

- **Interactive Lighting**

with
heart rate and circadian
monitoring

- **The Prototype**

of a
3D-printed cell fragment



Q&A

REFERENCES

9 REFERENCES

- Fig. 01, Rhino to Nano Banana Interior View, April 2026
<https://gemini.google.com/>
- Fig. 02, Blizzard at a station in Antarctica, April 2026
<https://www.antarctica.gov.au/about-antarctica/weather-and-climate/weather/>
- Fig. 03, Map Terrain Antarctica, April 2026
<https://news.osu.edu/a-terrain-map-that-shows-antarctica-in-stunning-detail/>
- Fig. 04, Polar night / Aurora australis on Antarctica, April 2026
<https://tinyurl.com/3fahry7b>
- Fig. 05, Daylight conditions, Troll station, April 2026
Diagram produced by the authors
- Fig. 06, Troll Research Station, Norsk Polarinstittutt, April 2026
<https://npolar.no/en/tone/>
- Fig. 07, Functional division, April 2026
Diagram produced by the authors
- Fig. 08, Fundamental needs, April 2026
Diagram produced by the authors
- Fig. 09, Project goals, April 2026
Diagram produced by the authors
- Fig. 10, 24h activity mapping, February 2026
Diagram produced by the authors
- Fig. 11, Concept: Cave, April 2026
Diagram produced by the authors
- Fig. 12, Exterior Vision, AI generated (Nano Banana), April 2026
<https://gemini.google.com/>
- Fig. 13, Voronoi principle, April 2026
Diagram produced by the authors
- Fig. 14, KUKA Robot Additive Manufacture
<https://www.kuka.com/it-it/settori/banca-dati-di-soluzioni/2025/09/caracol-case-study>
- Fig. 15, DATA output / Schematic Print Process Fragment, April 2026
Diagram produced by the authors
- Fig. 16, DATA output / Fused particle fabrication scheme, April 2026
Diagram produced by the authors
- Fig. 17, Data implementation steps, source: Python script PolarH10, April 2026
Diagram or script output produced by the authors
- Fig. 18, Daylight Conditions, Troll Station / Interior perspective, April 2026
Figure number used inconsistently in the presentation
- Fig. 19, Spatial configuration, April 2026
Diagram produced by the authors
- Fig. 20, Diagram form finding script, April 2026
Script output or diagram produced by the authors
- Fig. 21, Diagram form finding script, April 2026
Script output or diagram produced by the authors
- Fig. 22, Video form finding script, April 2026
Script output or diagram produced by the authors
- Fig. 23, Lower Floor Spatial Configura Script output produced by the authors
Script output or diagram produced by the authors
- Fig. 24, Upper Floor Spatial Configuration, April 2026
Diagram produced by the authors
- Fig. 25, Upper and lower floor plans, April 2026
Drawing produced by the authors
- Fig. 26, Longitudinal and cross sections, April 2026
Drawing produced by the authors
- Fig. 27, Voronoi typologies, April 2026
Diagram produced by the authors
- Fig. 28, GIFF folding cabinets, April 2026
Diagram or visualization produced by the authors
- Fig. 29, GIFF entrance cabinets, April 2026
Diagram or visualization produced by the authors
- Fig. 30, GIFF privacy niche, April 2026
Diagram or visualization produced by the authors
- Fig. 31, GIFF folding table, April 2026
Diagram or visualization produced by the authors
- Fig. 32, Lighting principle, April 2026
Diagram produced by the authors
- Fig. 33, Lighting principle, V-Ray Rendering, March 2026
Diagram produced by the authors
- Fig. 34, Equalizing heartrates, April 2026
Diagram produced by the authors
- Fig. 35, Lighting principle, April 2026
Diagram produced by the authors
- Fig. 36, CCT adjustment based on activity mapping, April 2026
Diagram produced by the authors
- Fig. 37, Lighting Script Grasshopper, Overview, April 2026
Script output produced by the authors
- Fig. 38, Lighting Script Grasshopper, Input Parameters, April 2026
Script output produced by the authors
- Fig. 39, Python Script HR Simulation, April 2026
Script output produced by the authors
- Fig. 40, Lighting Script Grasshopper, Proximity amplification, April 2026
Script output produced by the authors
- Fig. 41, Lighting Script Grasshopper, Calculation tolerance threshold, April 2026
Script output produced by the authors
- Fig. 42, Lighting Script Grasshopper, Heart Rate-Driven colour temperature adjustment, April 2026
Script output produced by the authors
- Fig. 43, Tamal-sen / colourTemperatureToRGB (GitHib), April 2026
External source or adapted reference
- Fig. 44, Overview Lighting Script Grasshopper, Visualisation (Chaos V-Ray & Rhino Preview), April 2026
Visualization produced by the authors
- Fig. 45, Chaos V-Ray & Rhino Preview, April 2026
Visualization produced by the authors
- Fig. 46, Lighting Script Grasshopper, Lighting script demonstration, April 2026
Script output produced by the authors