

# Social Housing | "Accretio"

Maximilian Michl | 4929640

2018 | 19



# Introduction | Contents

Flow Chart	01
Space Planning	02
Massing	12
Configuring	18
Forming	20
Result	25

## Design Prologue

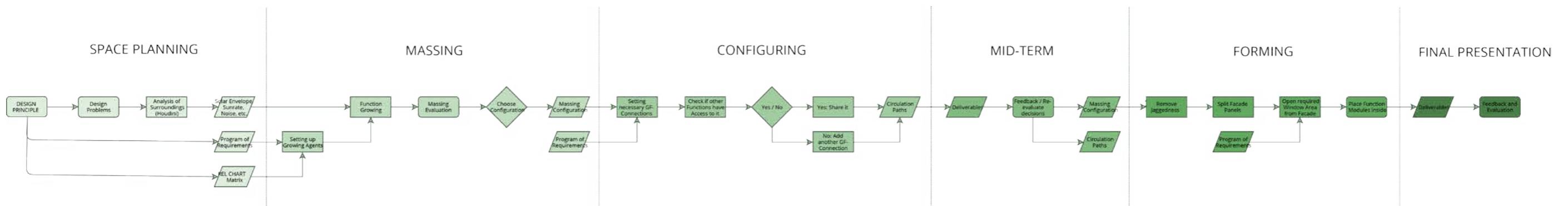
In the early stages, I kept on reencountering some main problems. How to open up the site and make the entrance more friendly? How to provide room to break out of your room? How to provide some privacy to the inhabitants with that many public functions? Through the use of the decided design principles many of those were mitigated if not completely resolved.

Another key aspect for me was to grow within the given site conditions making it a very specific design solution as well as allowing for the growth of a homogeneous community, spreading a positive environment inside the building.

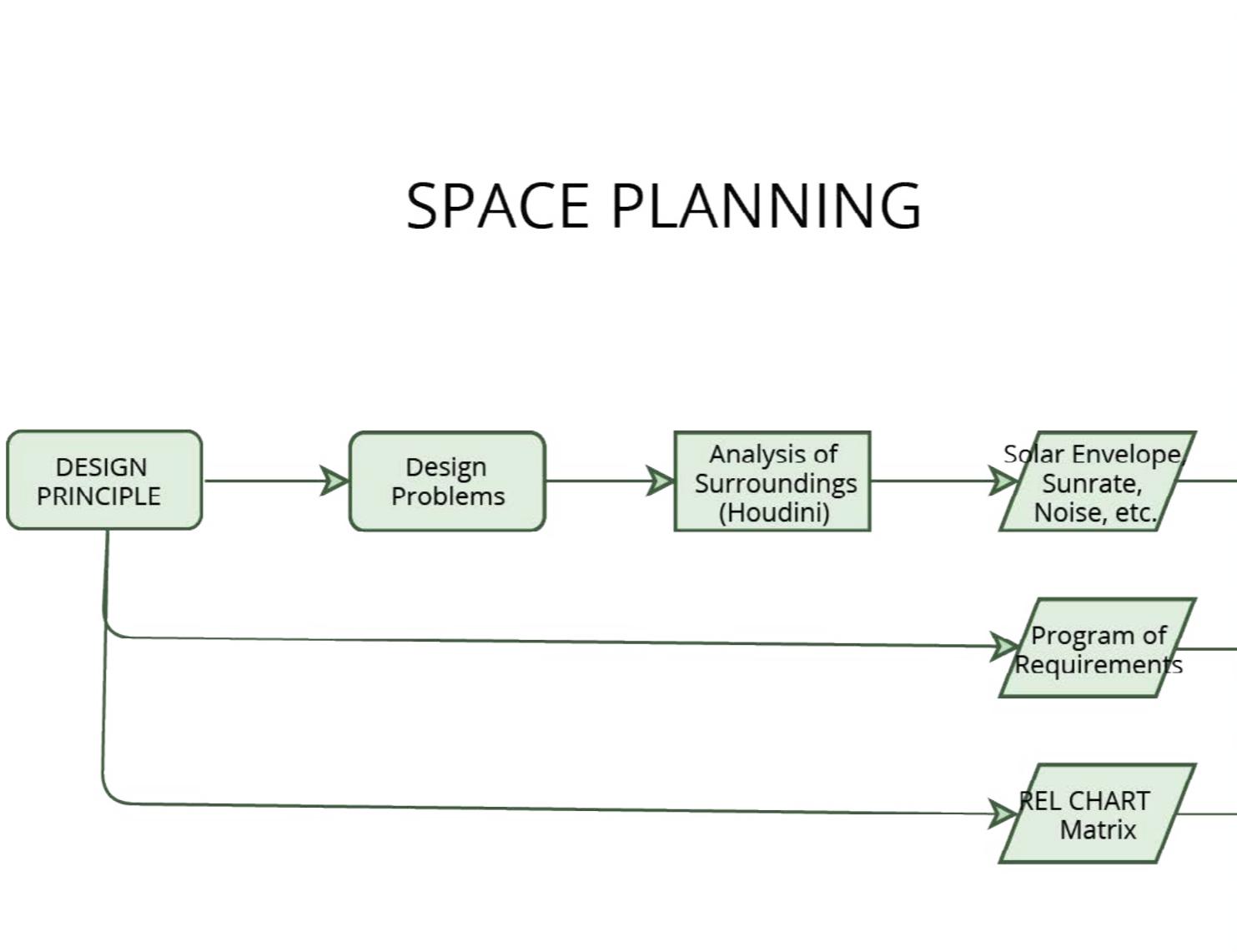
The growth of a building due to site conditions was especially interesting and therefore, became the main focus of my procedural modelling and coding. After a thorough analysis of the site and its external factors a two way optimized agent based growth was implemented. Functions move into an equilibrium between optimal conditions and relations between each other.

As a last step an algorithm was added to split up the façade and find best positions for the windows. The split is to a certain degree random, creating various variations throughout the panels.

# Flow Chart | Overview



## Flow Chart | Space Planning



## Space Planning | Site



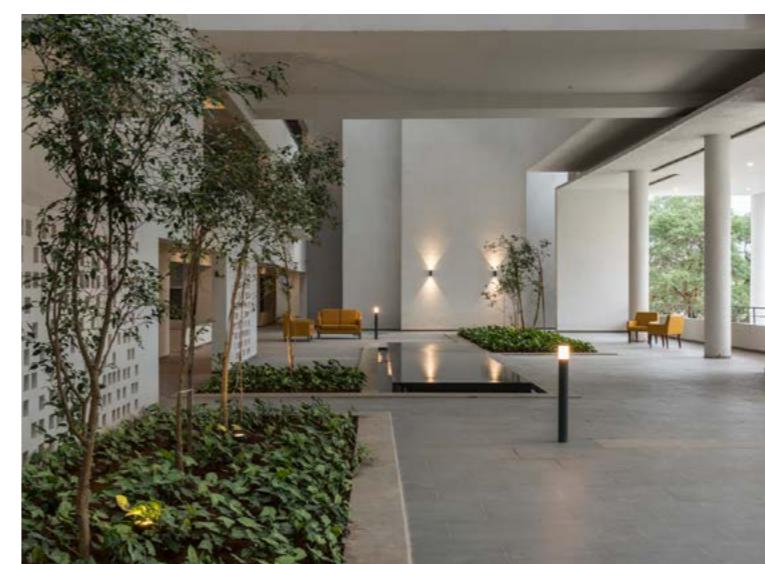
## Space Planning | Site



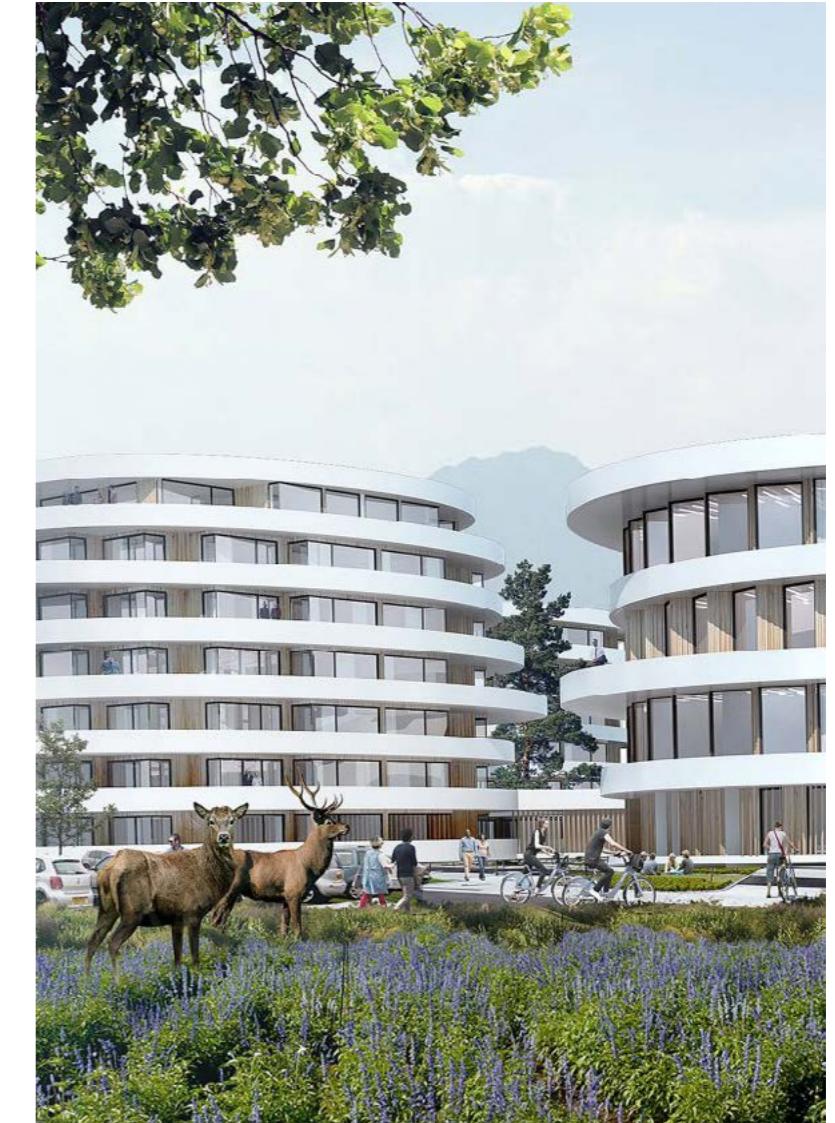
# Space Planning | Precedents



**Figure 1 |** Modular Housing, krft, Munich, Germany

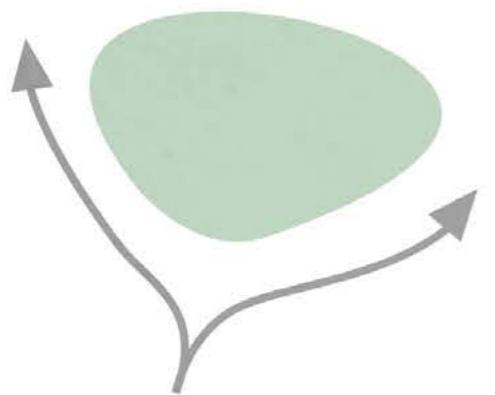


**Figure 2 |** Parkside Retirement Homes, Mindspace, India

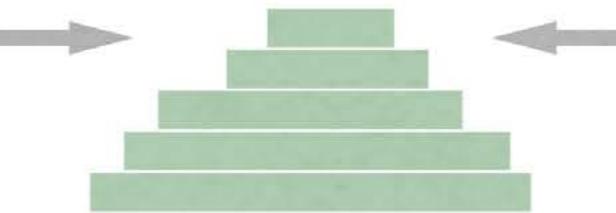


**Figure 3 |** Rondo, Atelier Thomas Pucher, Austria

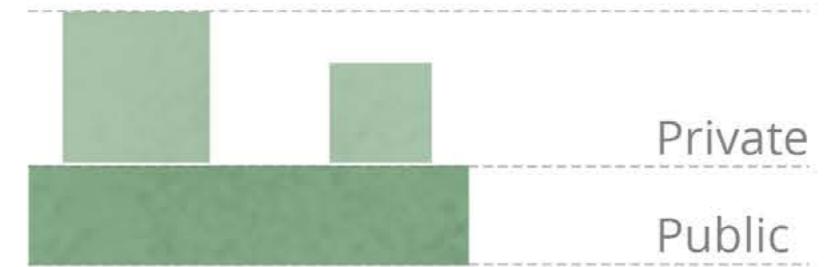
# Space Planning | Design Principles



**Design Principle** Fluid Movement



**Design Principle** Free up Top Space



**Design Principle** Zoning

# Space Planning | Relations

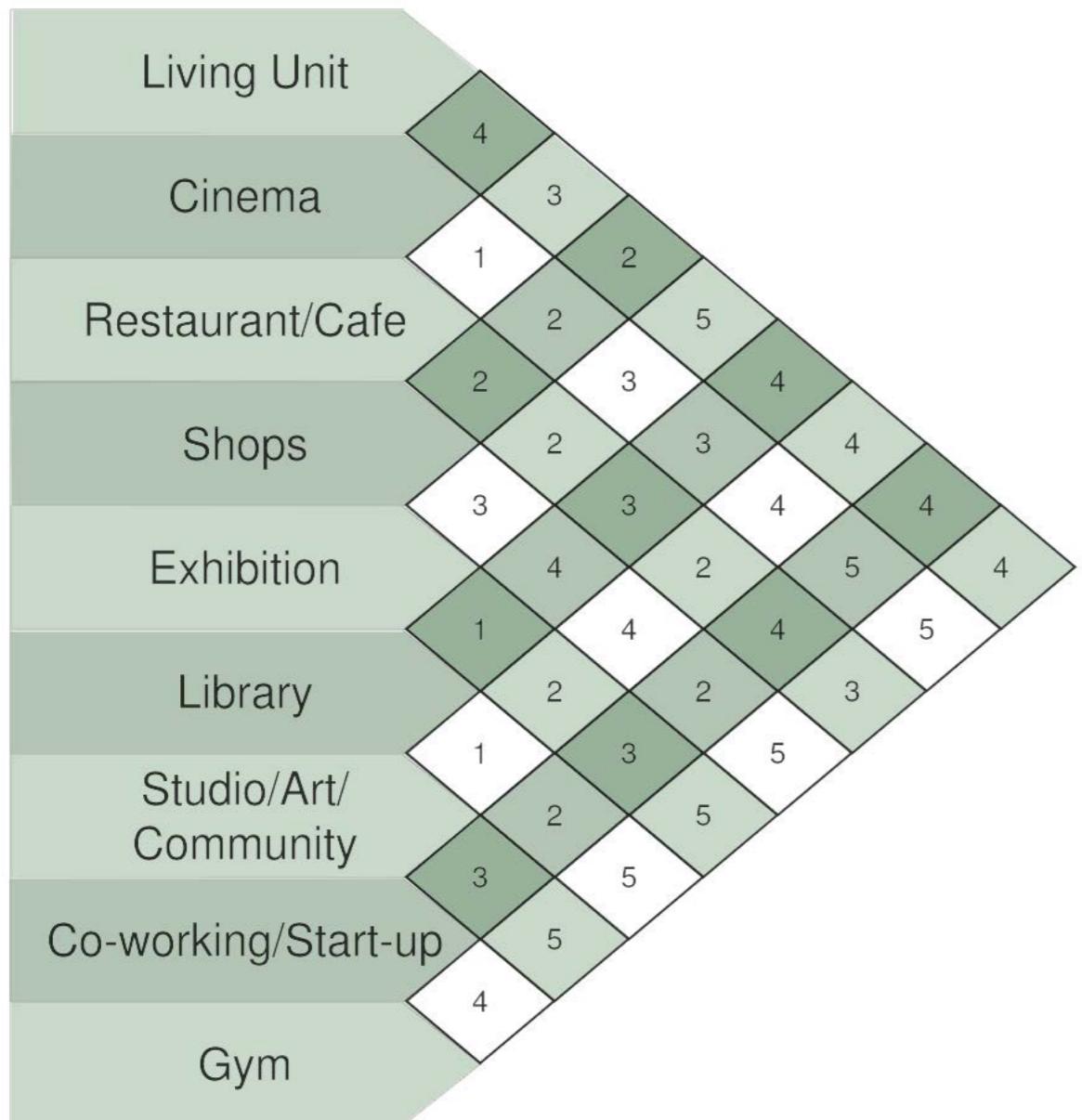


Figure 1 | REL Chart

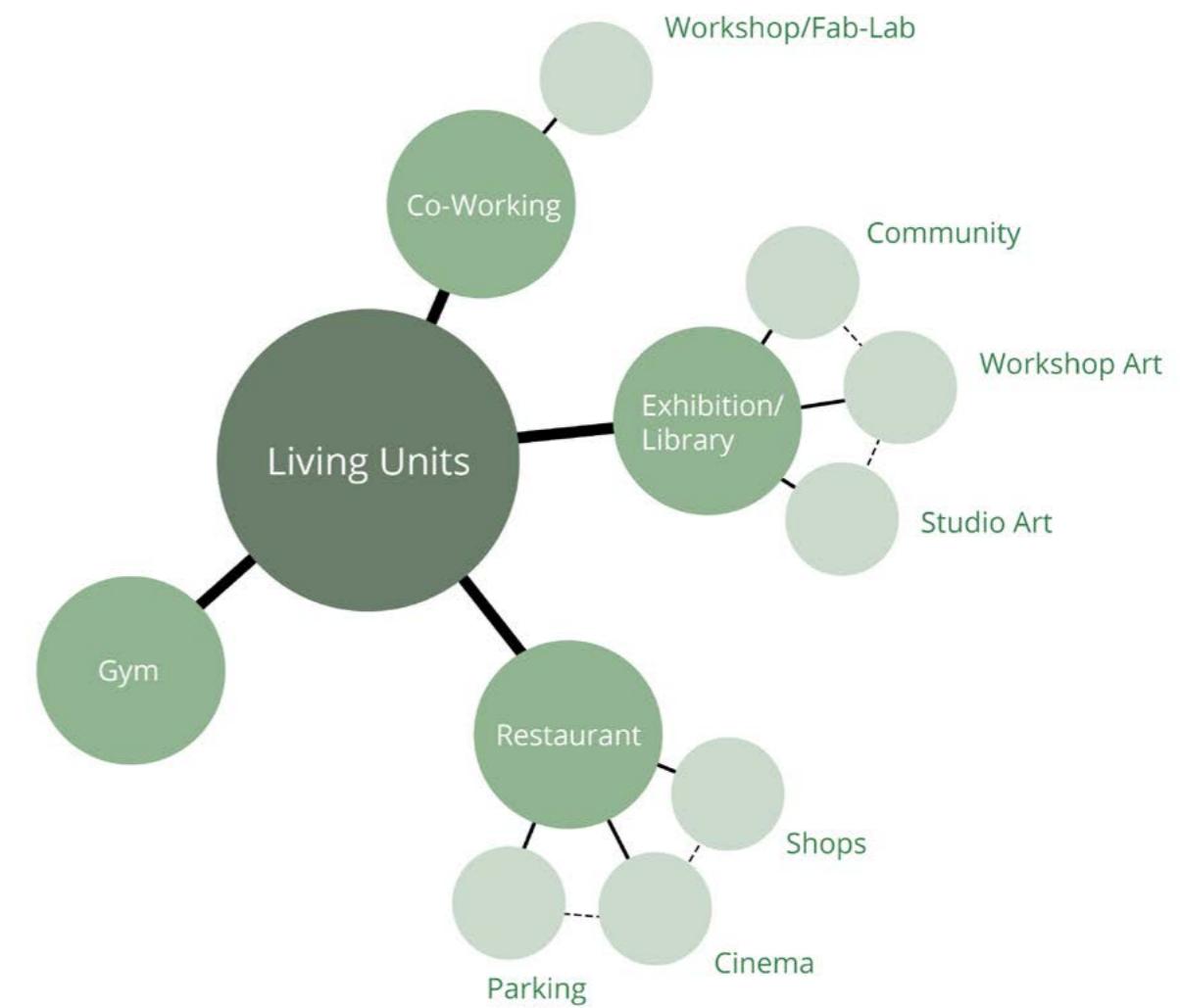
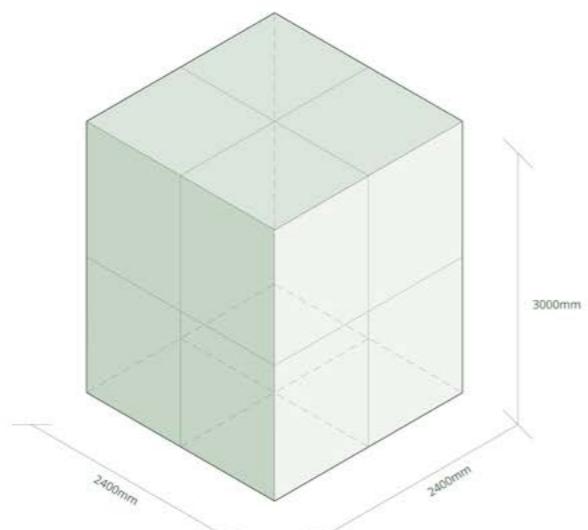
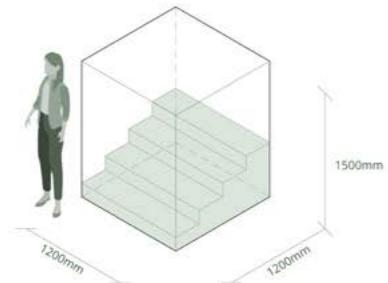


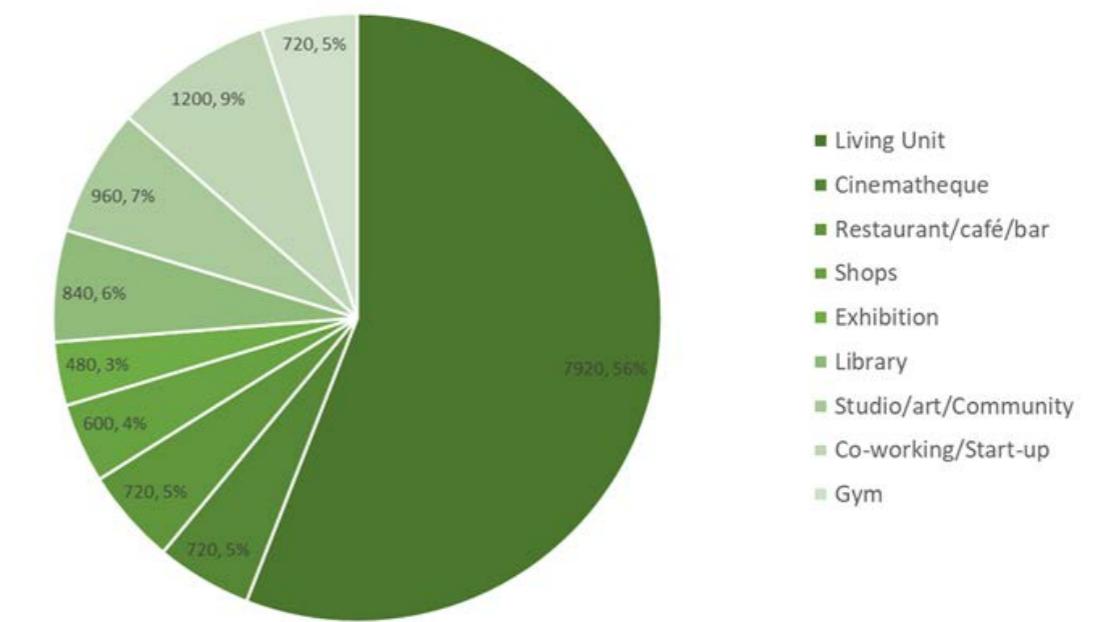
Figure 2 | Bubble Diagram

# Space Planning | Framework



**Figure 1 |**Voxel Size

Area Contribution of Functions

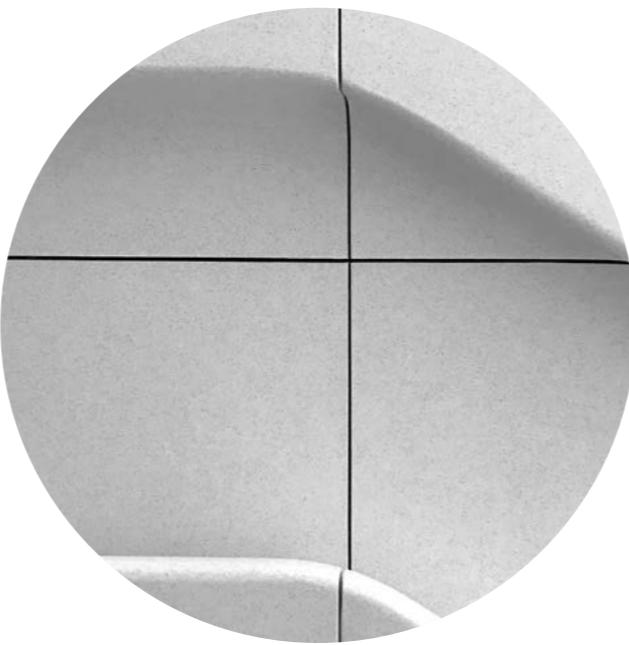


**Figure 2 |**Area Spread

## Space Planning | Materials



**Figure 1** | Zinc Panels (Facade)

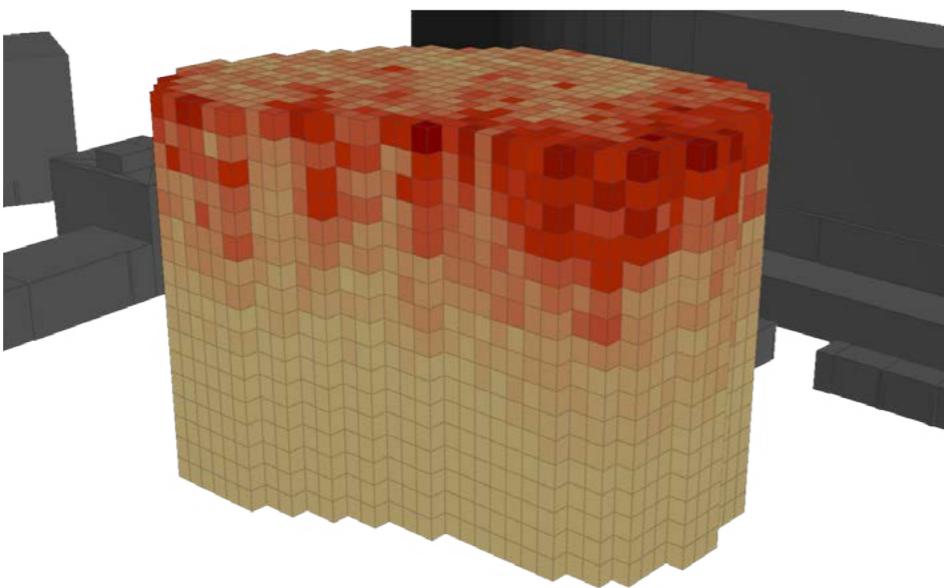


**Figure 2** | White GfRC (Guardrails)

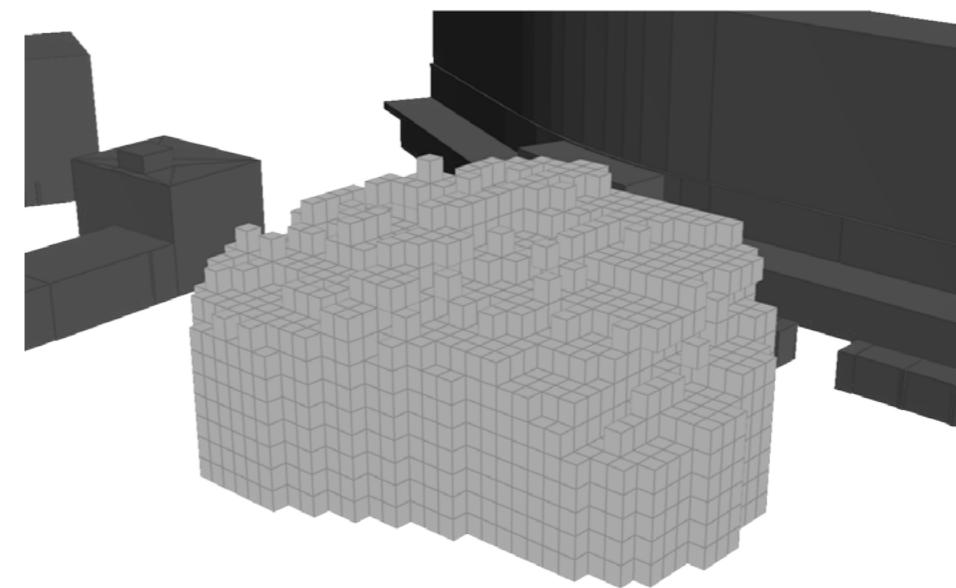


**Figure 3** | Vegetation (Steeped Terraces)

## Space Planning | Solar Envelope

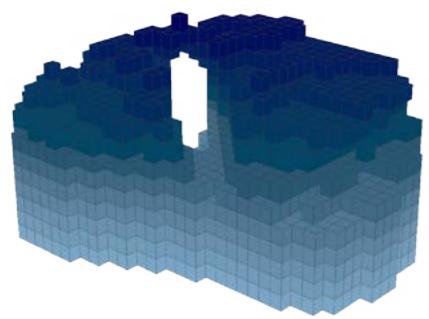


**Figure 1** | Hit Count

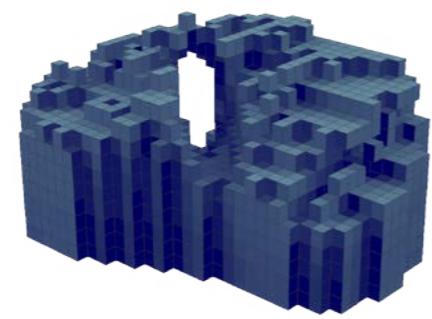


**Figure 2** | Solar Envelope

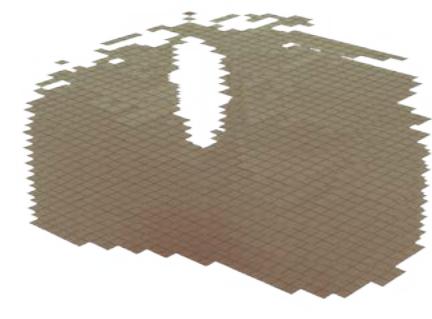
# Space Planning | Site Condition Evaluation (To be read with Pseudo Code 5)



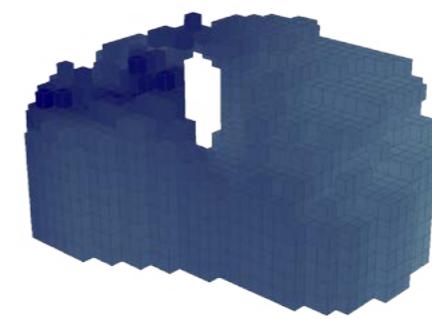
**Figure 1** | Floor Levels



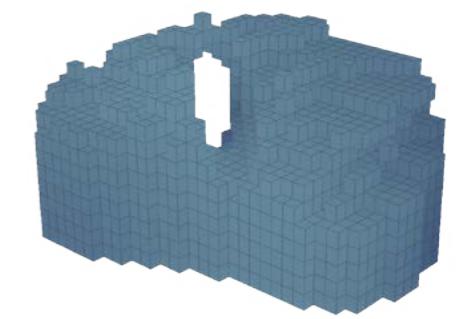
**Figure 2** | Sunlight Analysis



**Figure 3** | Daylight Spread

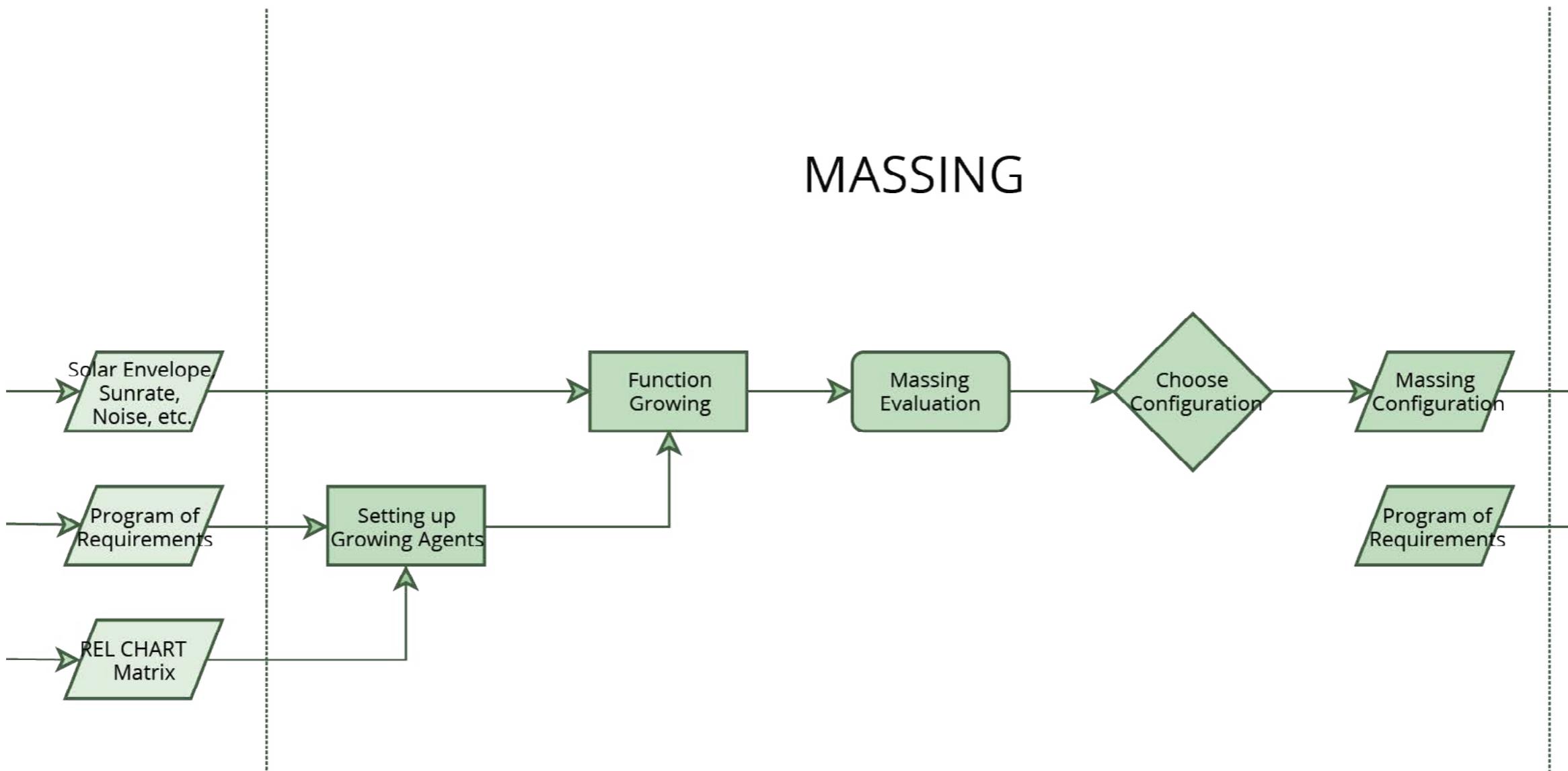


**Figure 4** | Connectivity



**Figure 5** | Noise

## Flow Chart | Massing



## Agent based growing

**For each** agent/function

Every fifty frames growth variable = 0

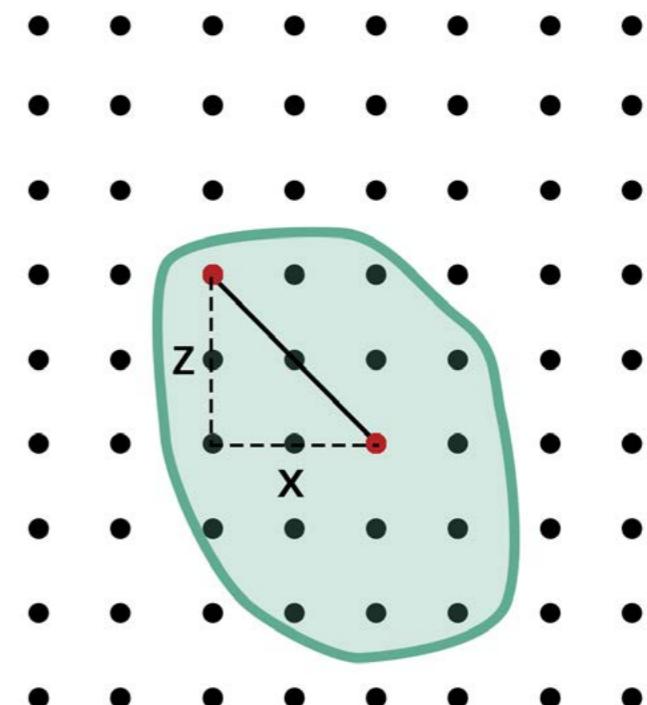
Read and **store** the agent attributes from its node and the growth variable

(**If** growth variable == 0) **For each** point in voxel point cloud

+ Calculate the centroid (loc) of function  $(\sum_{k=0}^n P.kxyz)/n$

+ **If** its x- or z-Distance >  $\sqrt{\text{area}/\text{floor}} * 0.8$  **OR** its y-Distance >  $(\text{floor\_height} * \text{floors})/2$

#**Remove** the voxel from the function



## Agent based growing

**For each** agent/function

Every fifty frames growth variable = 0

Read and **store** the agent attributes from its node and the growth variable

(**If** growth variable == 0) **For each** point in voxel point cloud

+ Calculate the centroid (loc) of function  $(\sum_{k=0}^n P.kxyz)/n$

+ **If** its x- or z-Distance >  $\sqrt{\text{area}/\text{floor}} * 0.8$  **OR** its y-Distance >  $(\text{floor\_height} * \text{floors})/2$

#**Remove** the voxel from the function

+ **Else; For each** of its neighbours :

# Read and **store** the relevant point attributes stored on it, as variables

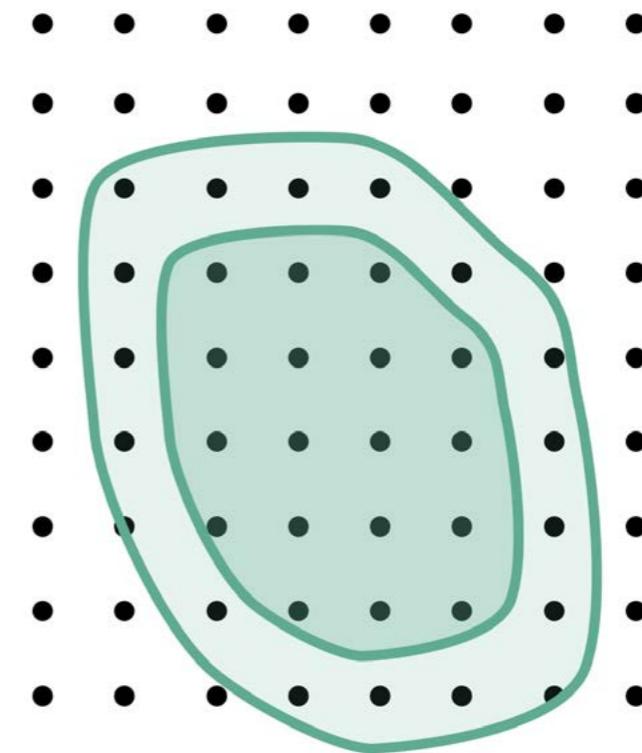
# **If** the point is within specified floor\_range; func\_id == 0; enough connectivity and day light as well as acceptable noise level

- **Calculate** the performance (**weighted product**)

- **If** performance > stored performance **AND** it has not been taken by

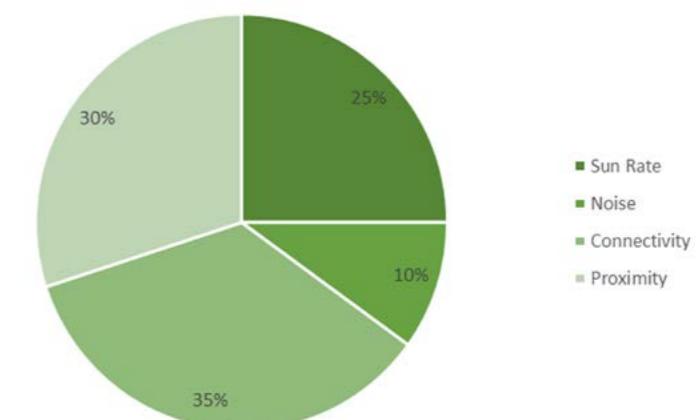
this agent before **AND** the function still needs more voxel

+**Overwrite** previous point\_id & performance



**Weighted Product | Example Gym**

$$P(A_K) = \prod_{j=1}^n (a_{Kj})^{w_j}, \text{ for } K = 1, 2, 3, \dots, m.$$



## Agent based growing

**For each** agent/function

Every fifty frames growth variable = 0

Read and **store** the agent attributes from its node and the growth variable

(**If** growth variable == 0) **For each** point in voxel point cloud

+ Calculate the centroid (loc) of function  $(\sum_{k=0}^n P.xyz)/n$

+ **If** its x- or z-Distance >  $\sqrt{\text{area}/\text{floor}} * 0.8$  **OR** its y-Distance >  $(\text{floor\_height} * \text{floors})/2$

#**Remove** the voxel from the function

+ **Else; For each** of its neighbours :

# Read and **store** the relevant point attributes stored on it, as variables

# **If** the point is within specified floor\_range; func\_id == 0; enough connectivity and day light as well as acceptable noise level

- **Calculate** the performance (**weighted product**)

- **If** performance > stored performance **AND** it has not been taken by

this agent before **AND** the function still needs more voxel

+ **Overwrite** previous point\_id & performance

+ **If** Voxel amount is reached **AND** growth variable is 0

# **Check** the voxel performance furthest away from the centroid compared to the ones it could grow to

# **If** there is a better performing one

- **Overwrite** the voxel with the new one

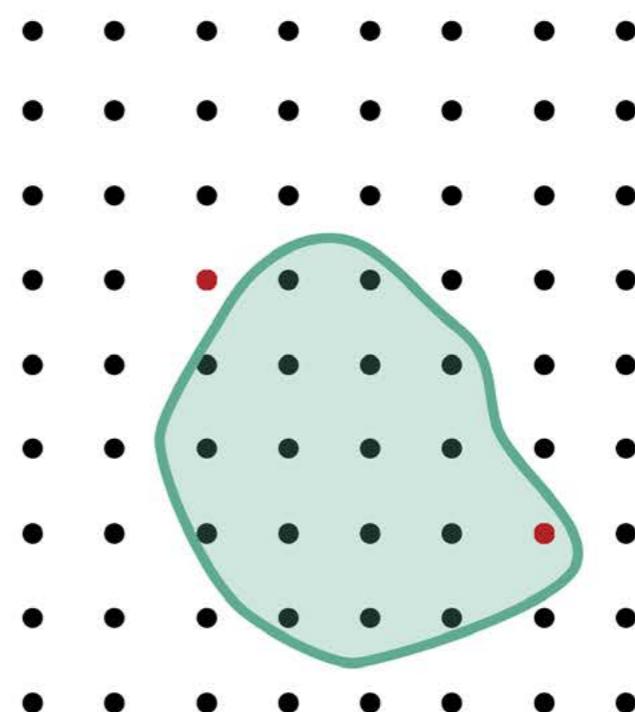
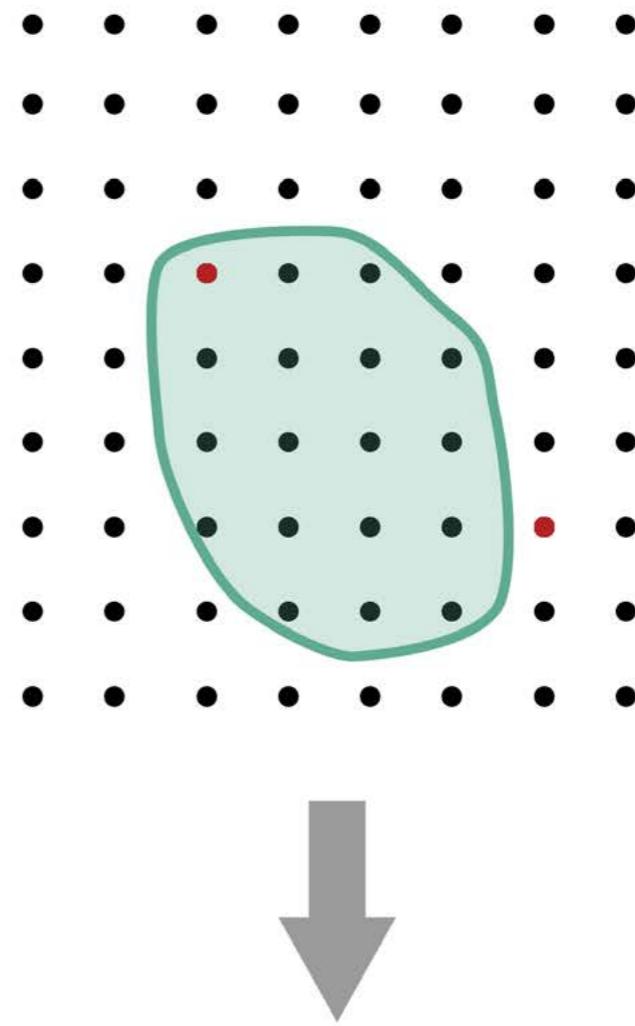
# **Else**

- Set growth variable to 1

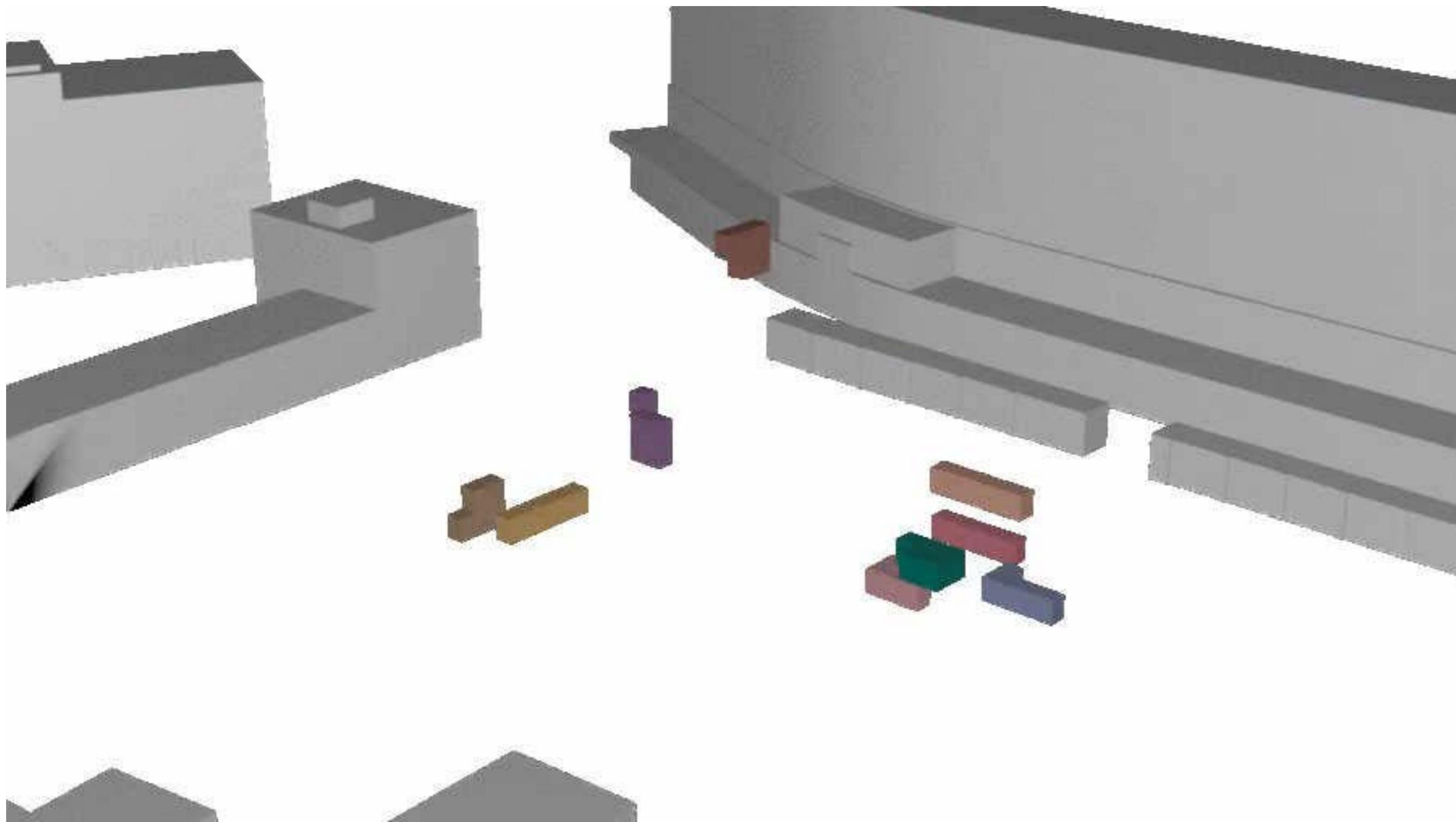
+ **If** id < 5000000 (Check for if it does not find a point)

#**Set** func\_id and group of that voxel to corresponding function

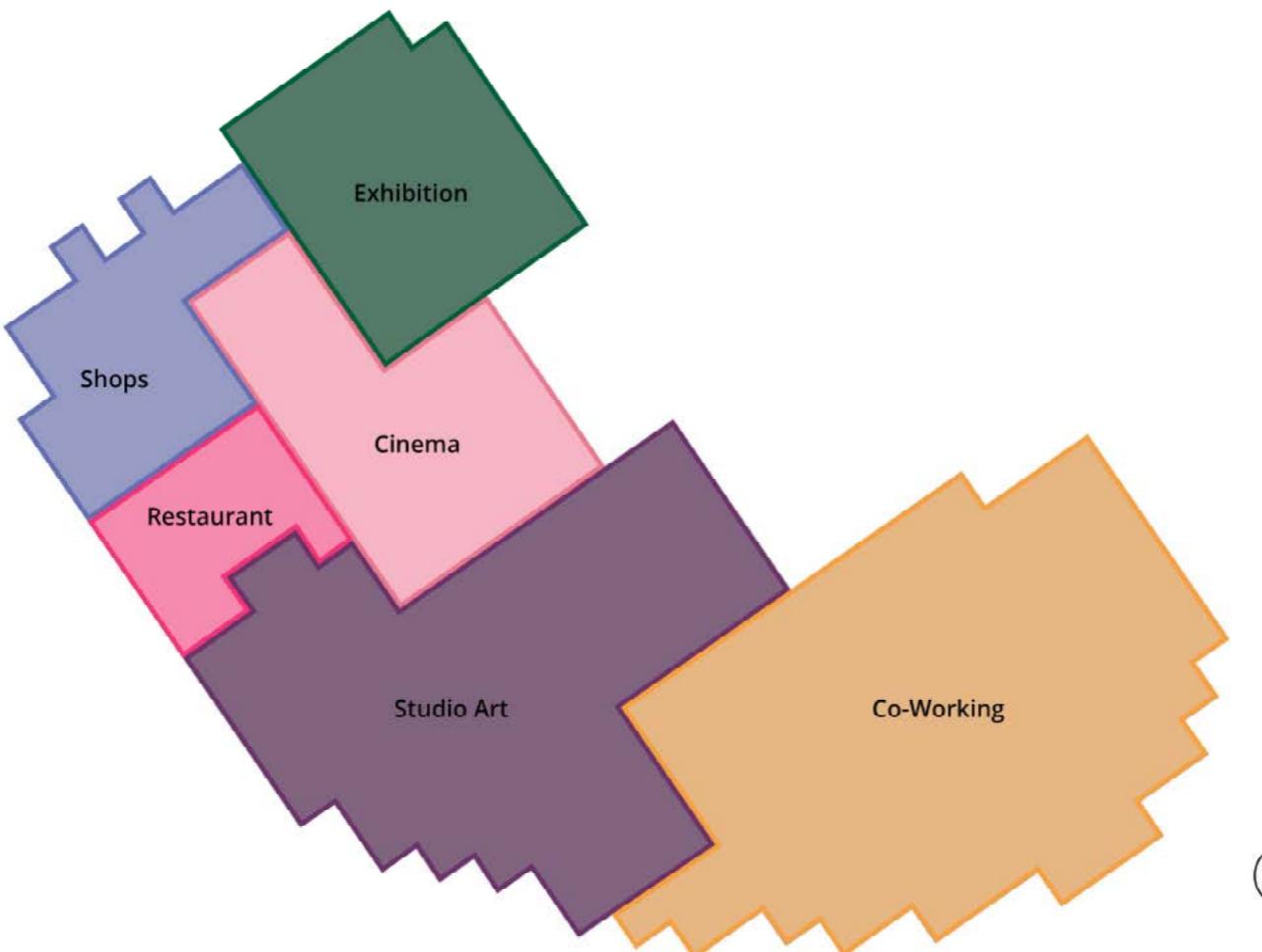
#**Add** 1 to its occupation attribute



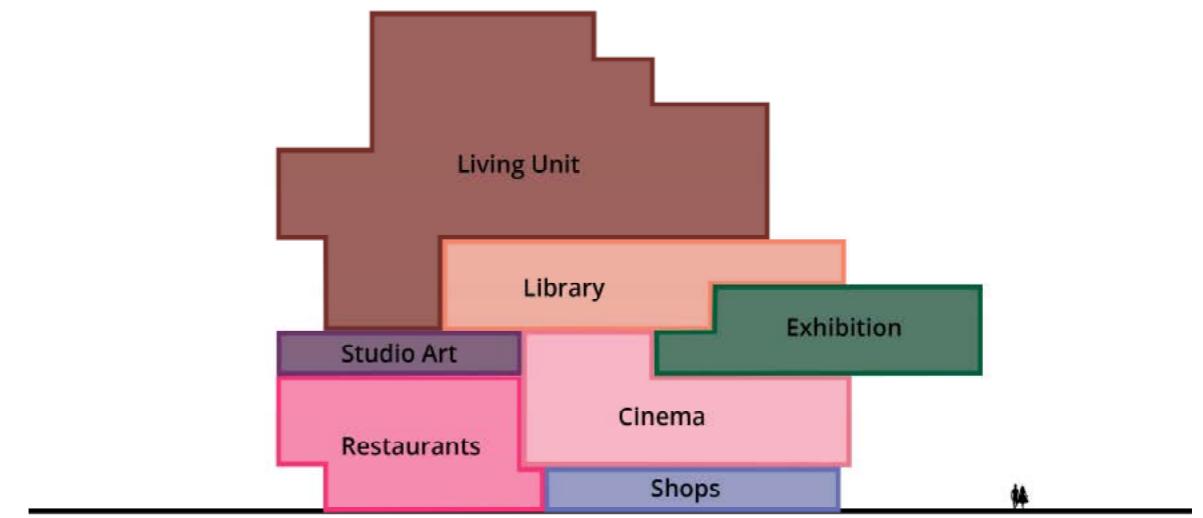
## Massing | Agent Based Growth



# Massing | Plan and Section

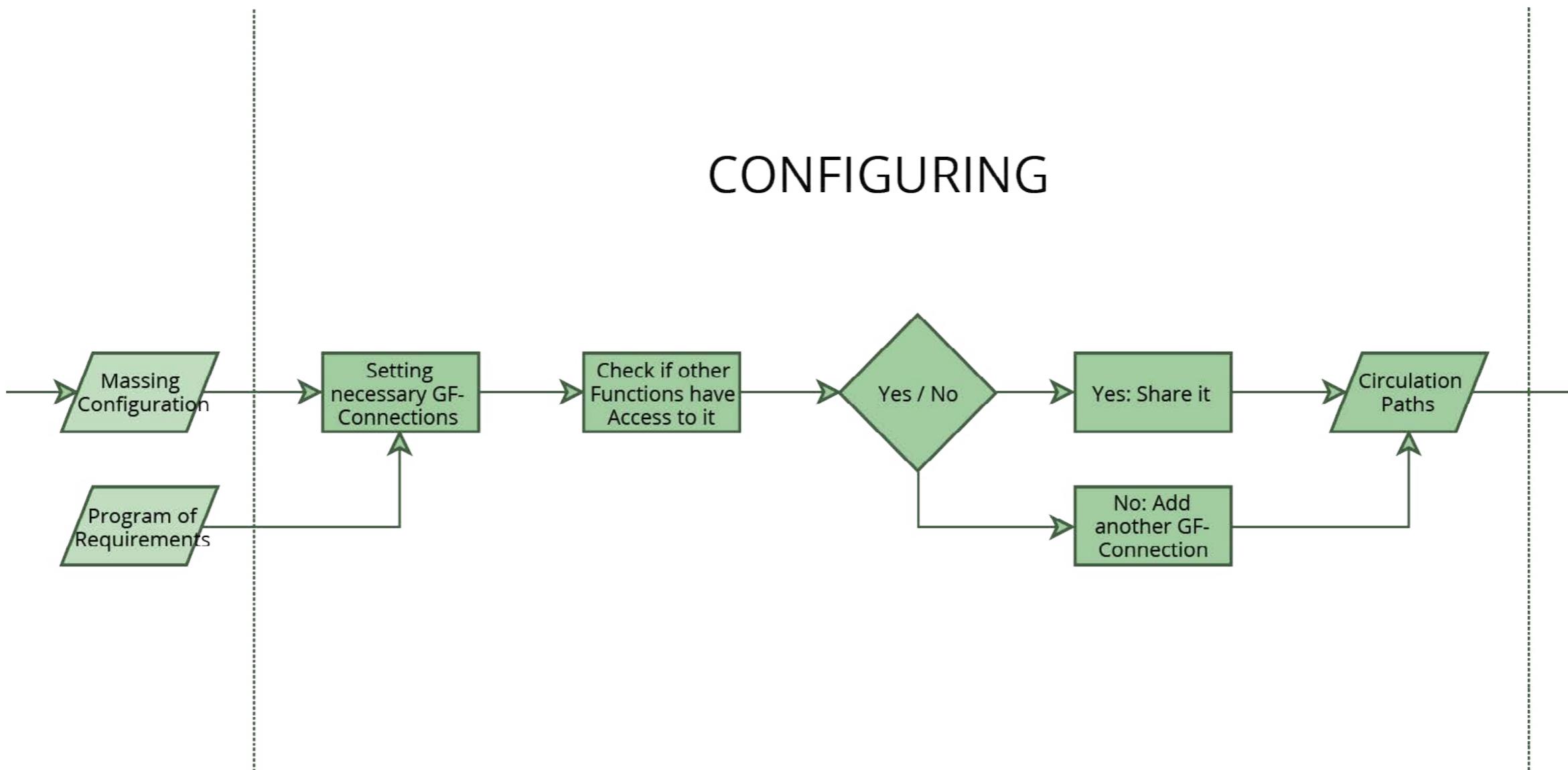


Plan | 1:500

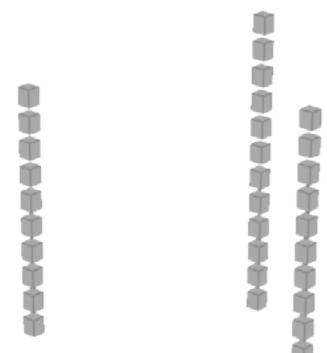


Section | 1:500

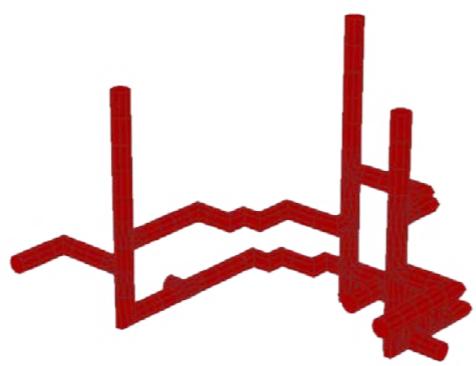
## Flow Chart | Configuring



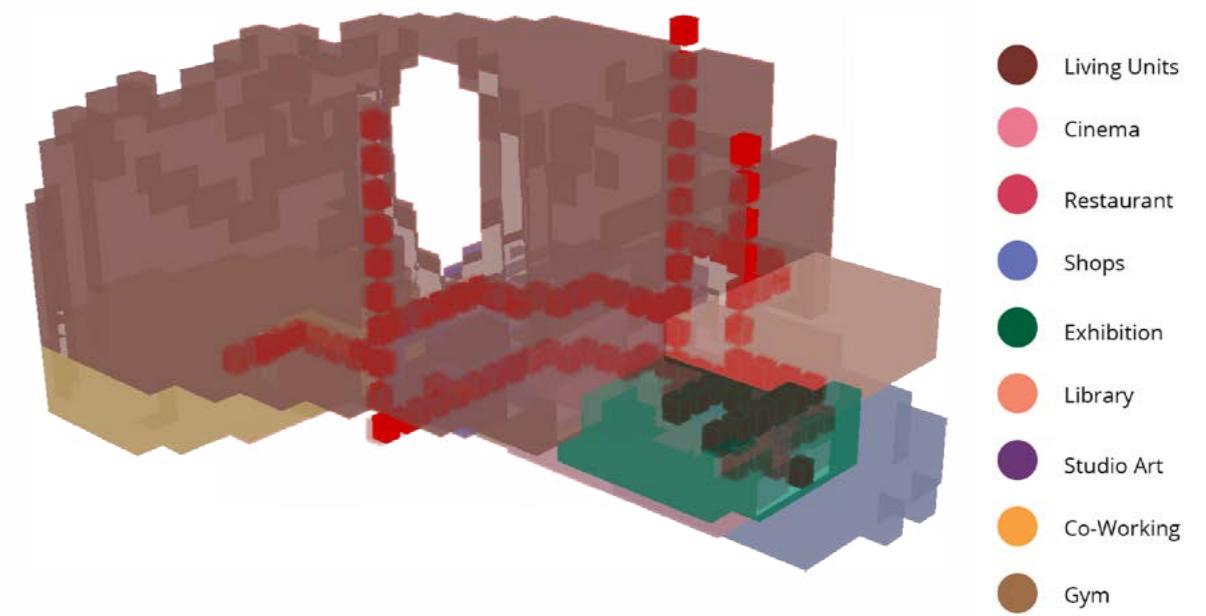
# Configuring | Path Finding



**Figure 1** | Circulation Shafts

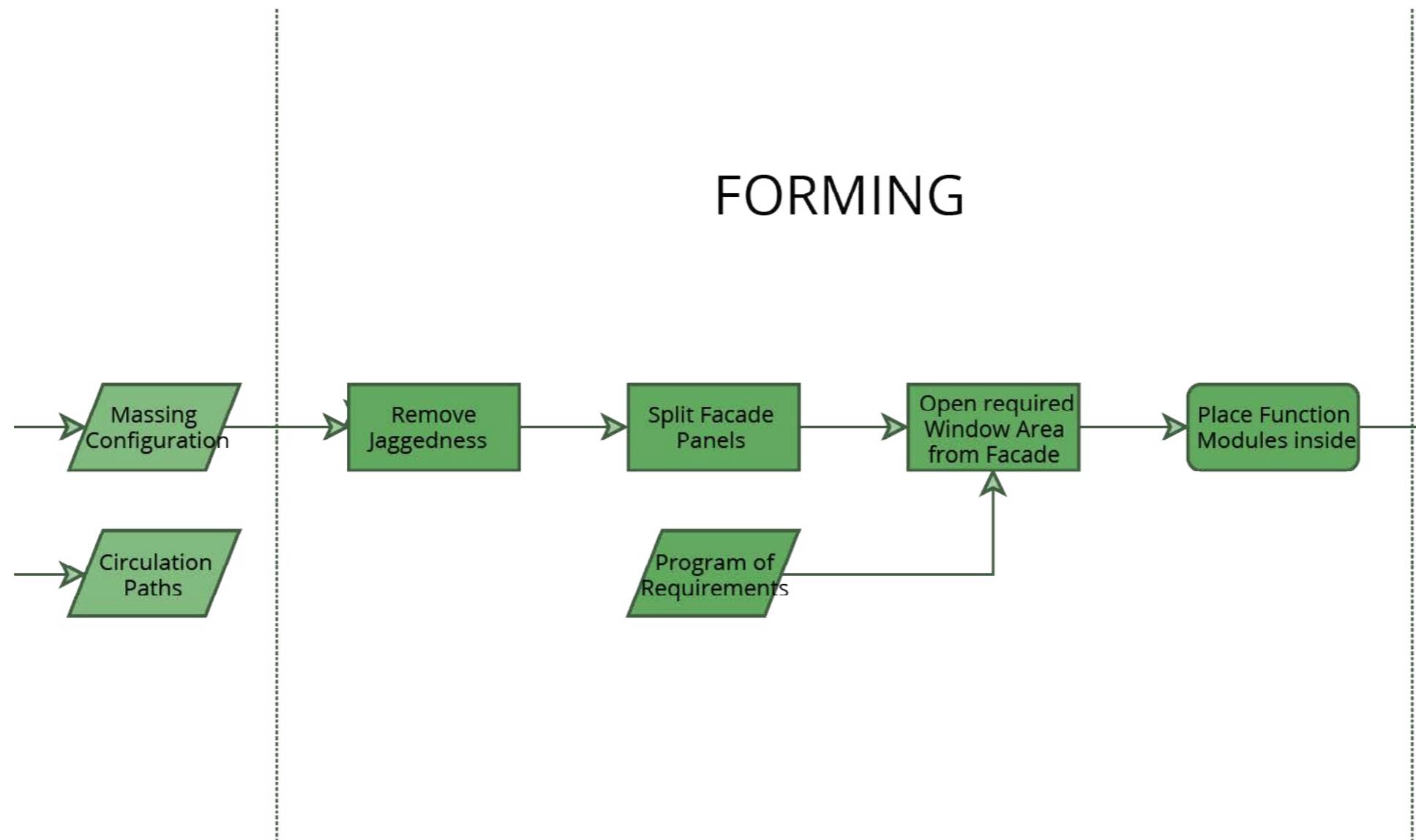


**Figure 2** | Pathways

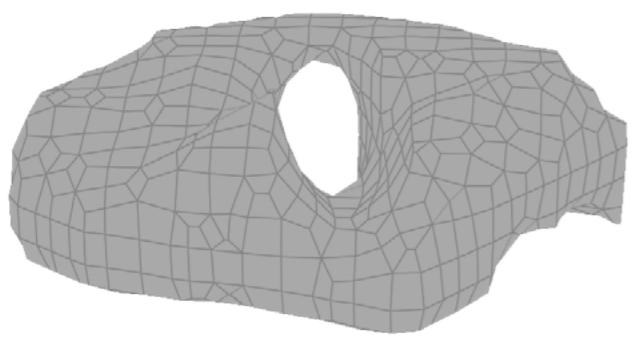


**Figure 3** | Internal Circulation

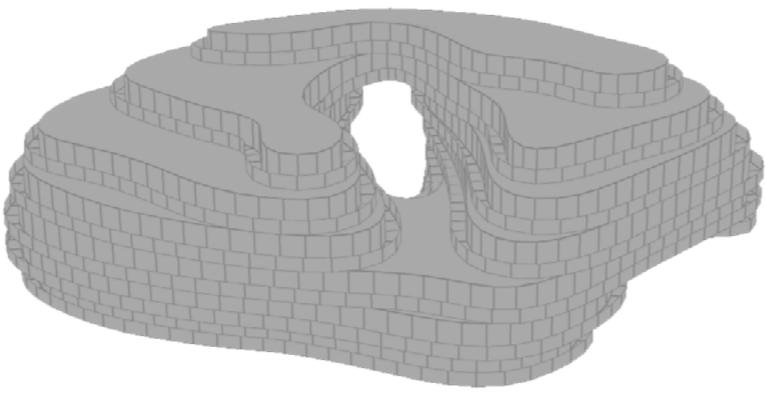
## Flow Chart | Forming



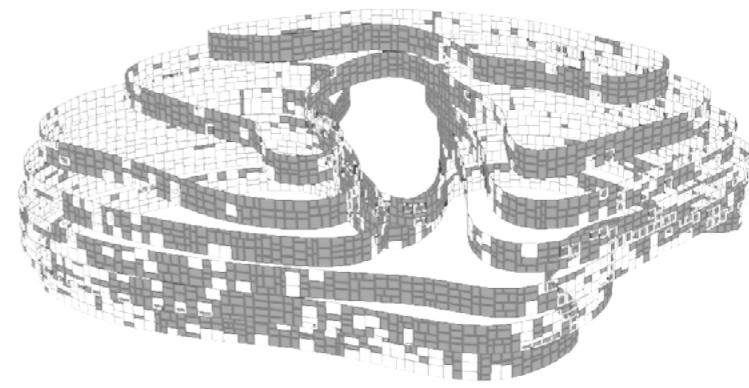
## Forming | Steps



**Figure 1** | VDB

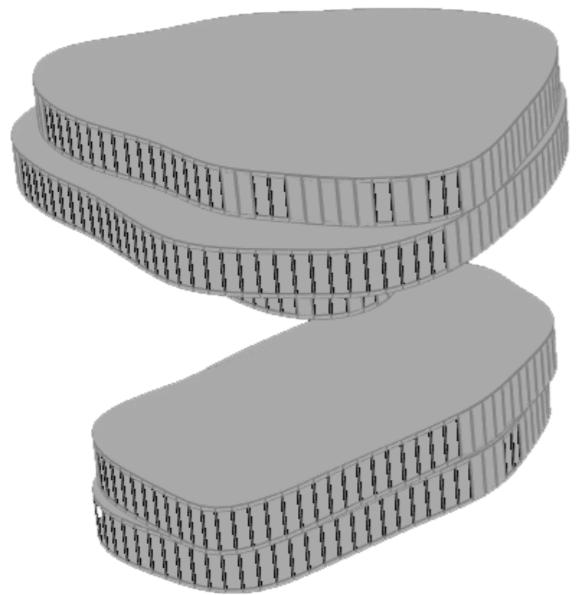


**Figure 2** | Face Extrusion

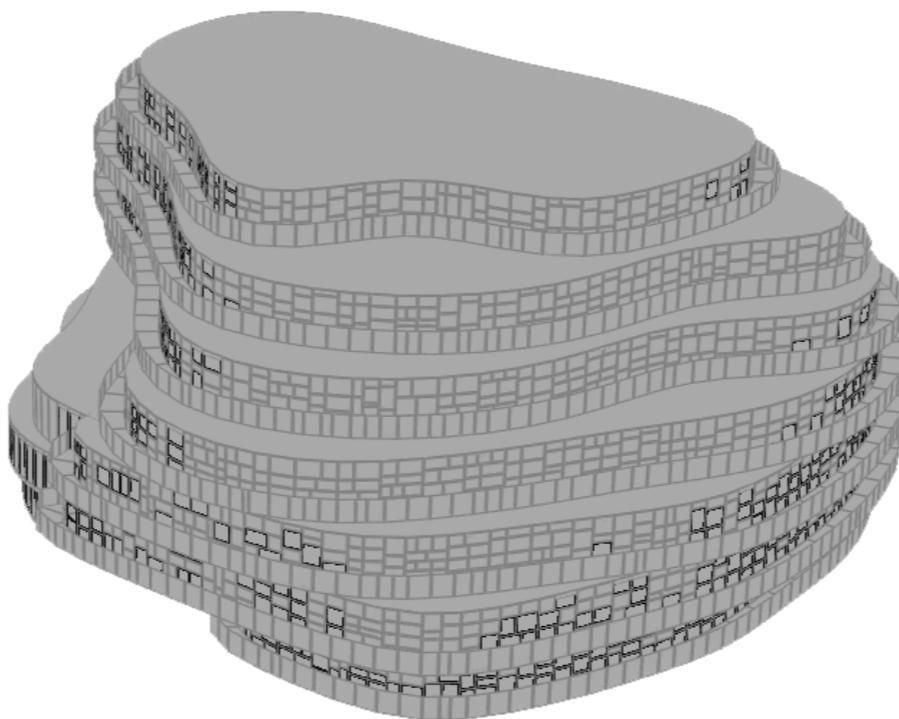


**Figure 3** | Facade Tiling and Windows

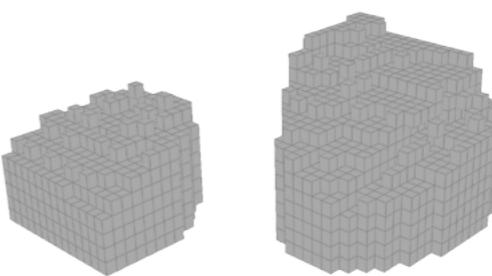
## Forming | Variations



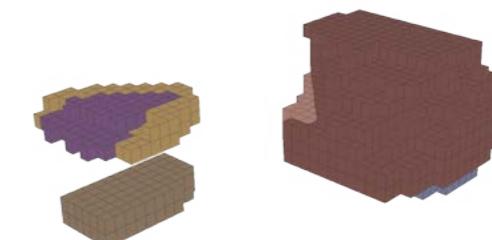
**Figure 1 |**Final Facade



**Figure 2 |**Box Volume

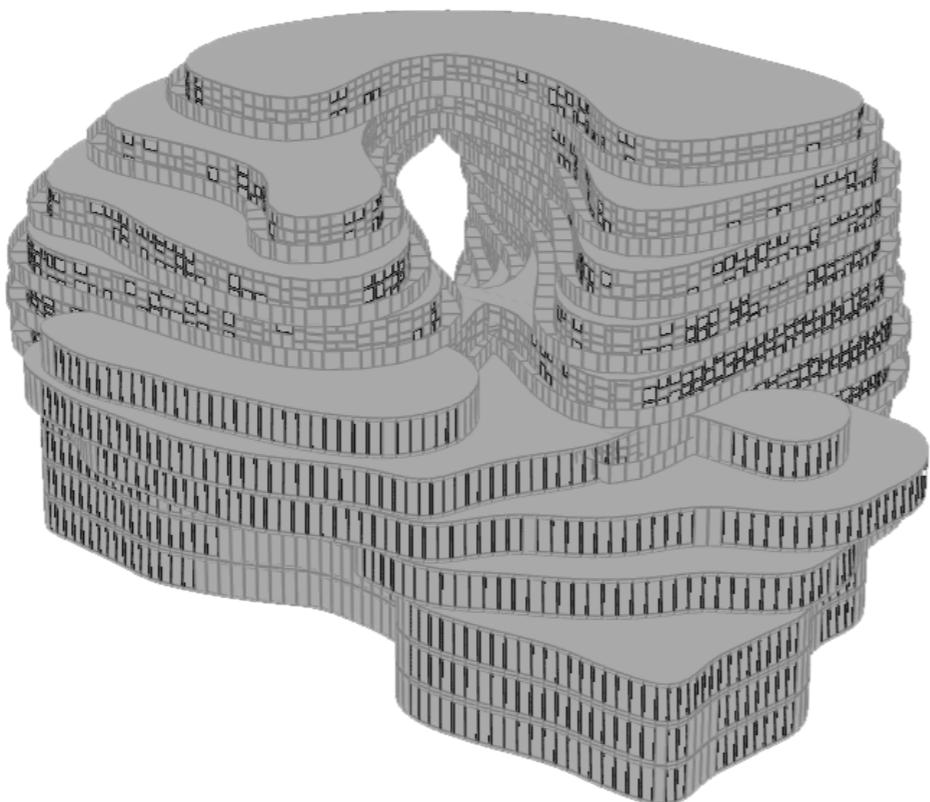


**Figure 3 |**Solar Envelop

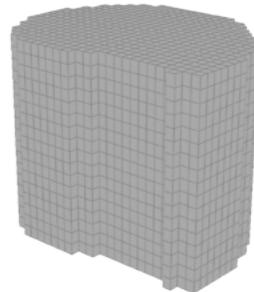


**Figure 4 |**Massing

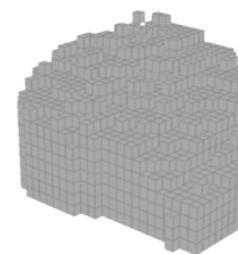
## Forming | Variations



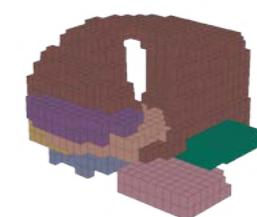
**Figure 1 |**Final Facade



**Figure 2 |**Box Volume

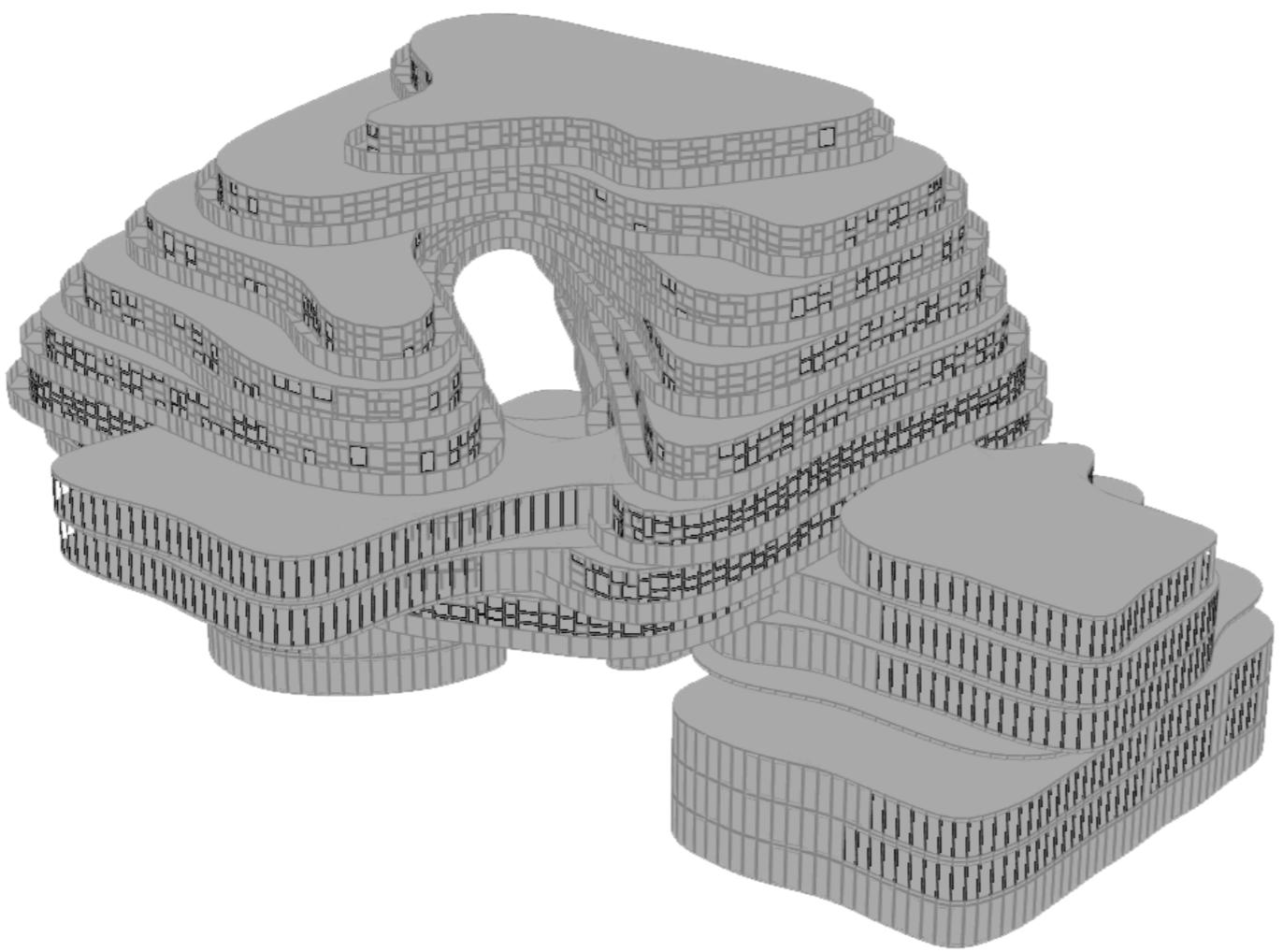


**Figure 3 |**Solar Envelop

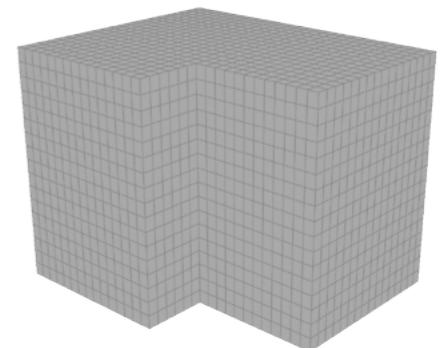


**Figure 4 |**Massing

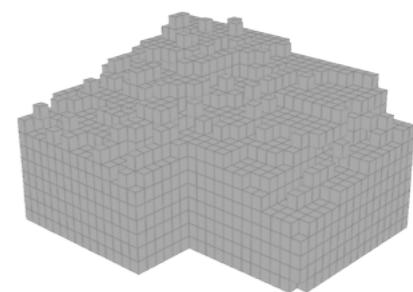
## Forming | Variations



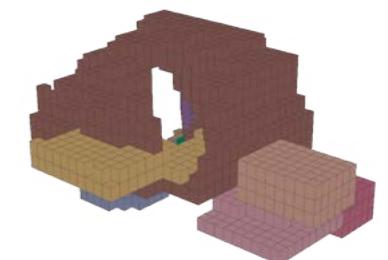
**Figure 1 |**Final Facade



**Figure 2 |**Box Volume



**Figure 3 |**Solar Envelop



**Figure 4 |**Massing

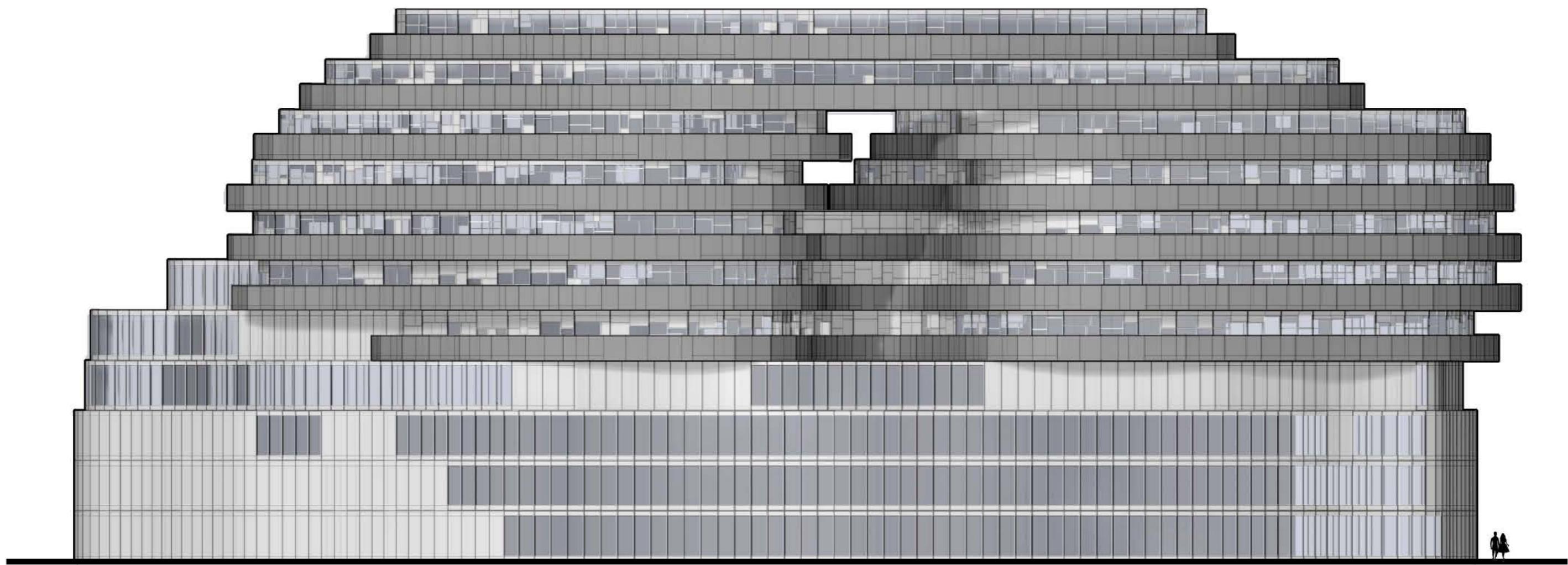
## Result | Urban Plan



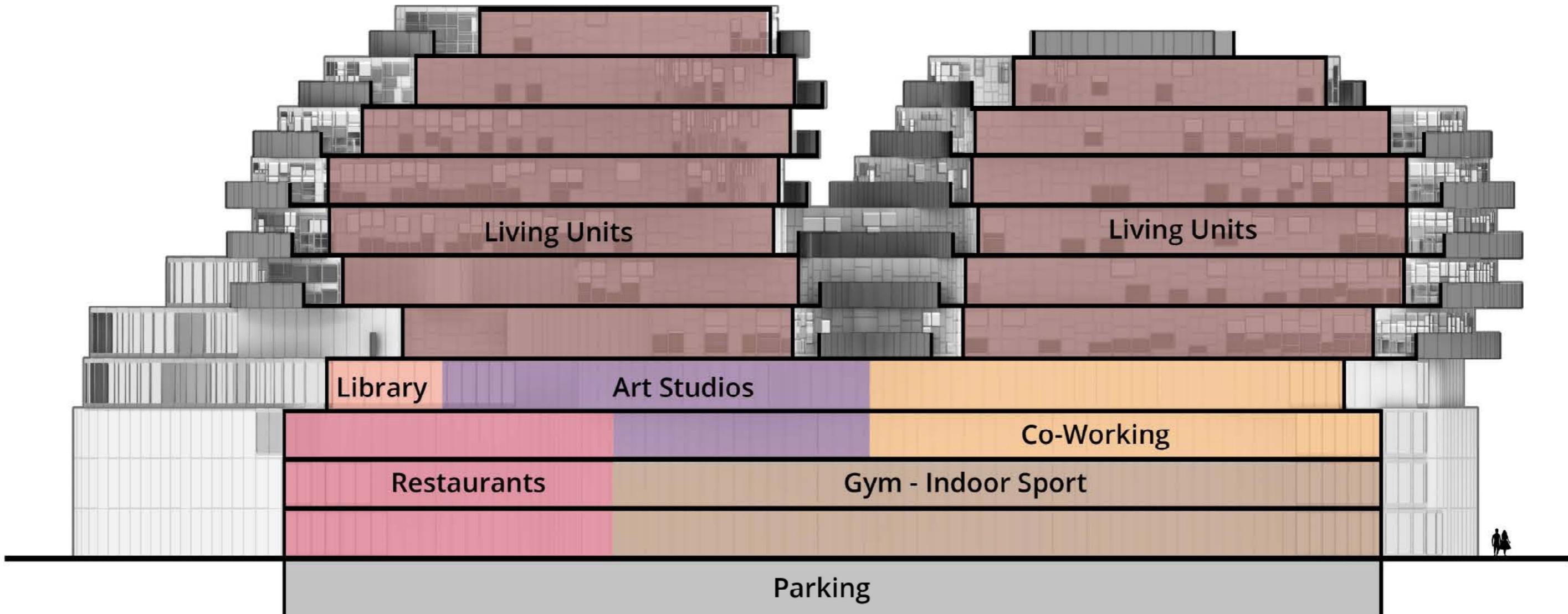
# Result | 7F Floor Plan



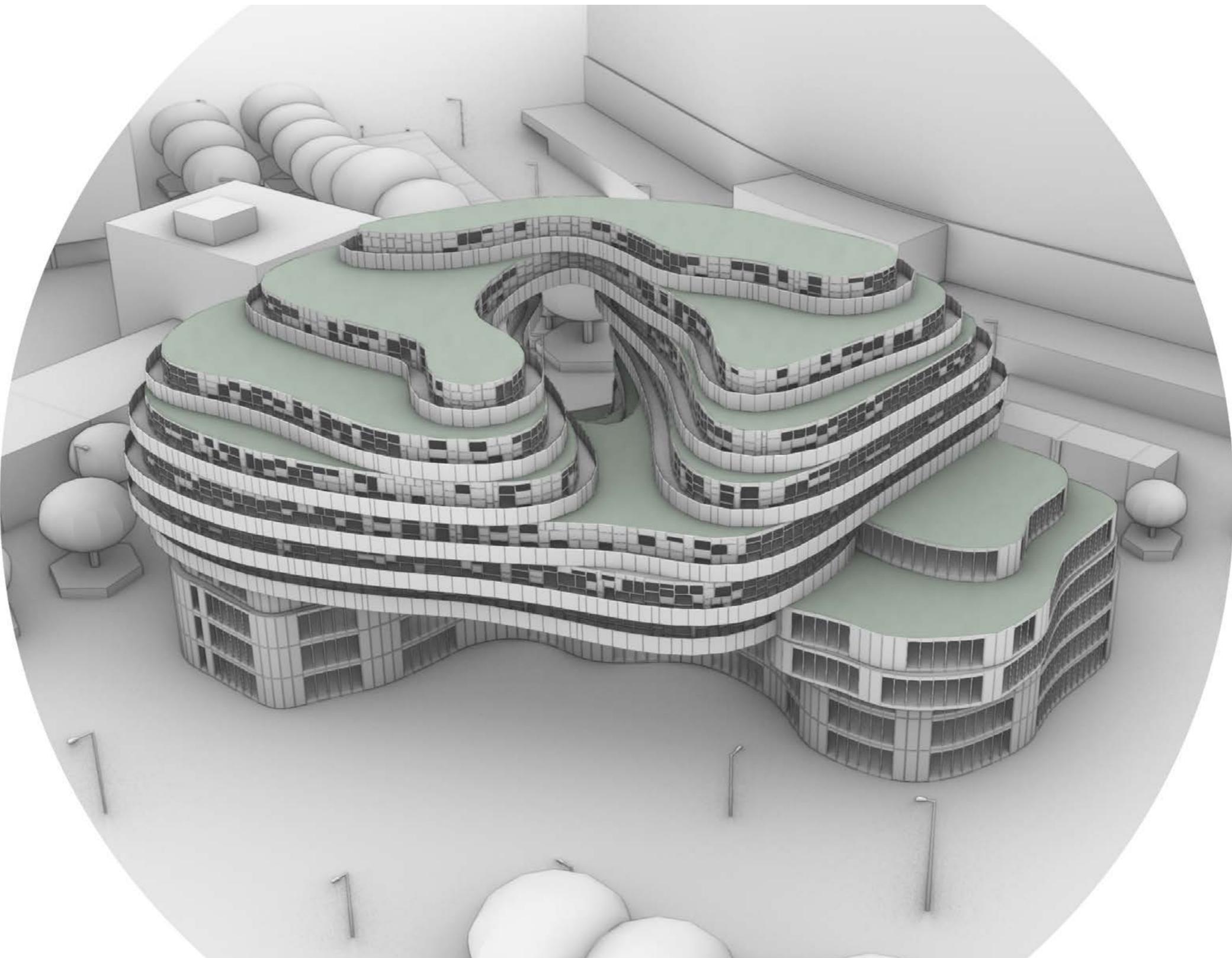
## Result | South Facade



## Result | Long Section AA'



## Result | Perspective



## Result | Perspective



## Result | Reflection

As this minor has a completely different agenda and focus compared to most studios at any other university, I was immediately drawn to it. I started with a lot of ideas on what I could achieve within the project and time space, however, after looking back on it now I might have had too lofty goals to achieve. Some of that might be due the unfortunate drop out of both of my group members, nonetheless, I would have liked to give the design part more time and overall achieve a more complete and integrated solution.

Especially in the beginning I felt like I was straying a bit of the path I wanted the project to go as there were many new aspects for me. Rekindling my lost programming knowledge, after a long time working in a group again as well as balancing between two courses and three coding languages. After splitting the different agendas up between us and focusing more on Houdini I slowly found my way around all of it.

Houdini in itself was very fun to use and I am glad I had this chance to start learning how to use it, as well as the whole idea behind the modelling in software's as Houdini or Grasshopper. I hope that all of this knowledge will help me along the way of my architecture career seeing as there are not that many architects yet, who can work with parametric decision making.

If I had to say what the most positive thing of me doing the project on my own was, I would say knowing the in and outs of every single detail of my code and design compared to others who might be a bit more focused on one part of this course. Faced with similar desperate situations, I think I would be able to hold my composure better and am, therefore, happy that this has happened rather now than sometime later down the line.

I am truly happy about all of the different techniques and concepts I learned during this minor and will try my best to further expand my knowledge on this part of design. Furthermore, it will be especially interesting how to implement it during the course of a regular design project which is not completely focused on computational design as it was this time. Ideally, the design would stay the main focus and these skills could act more as a tool and foil for achieving the best possible outcome. And that will also be the way forward for me in the next weeks and months.

