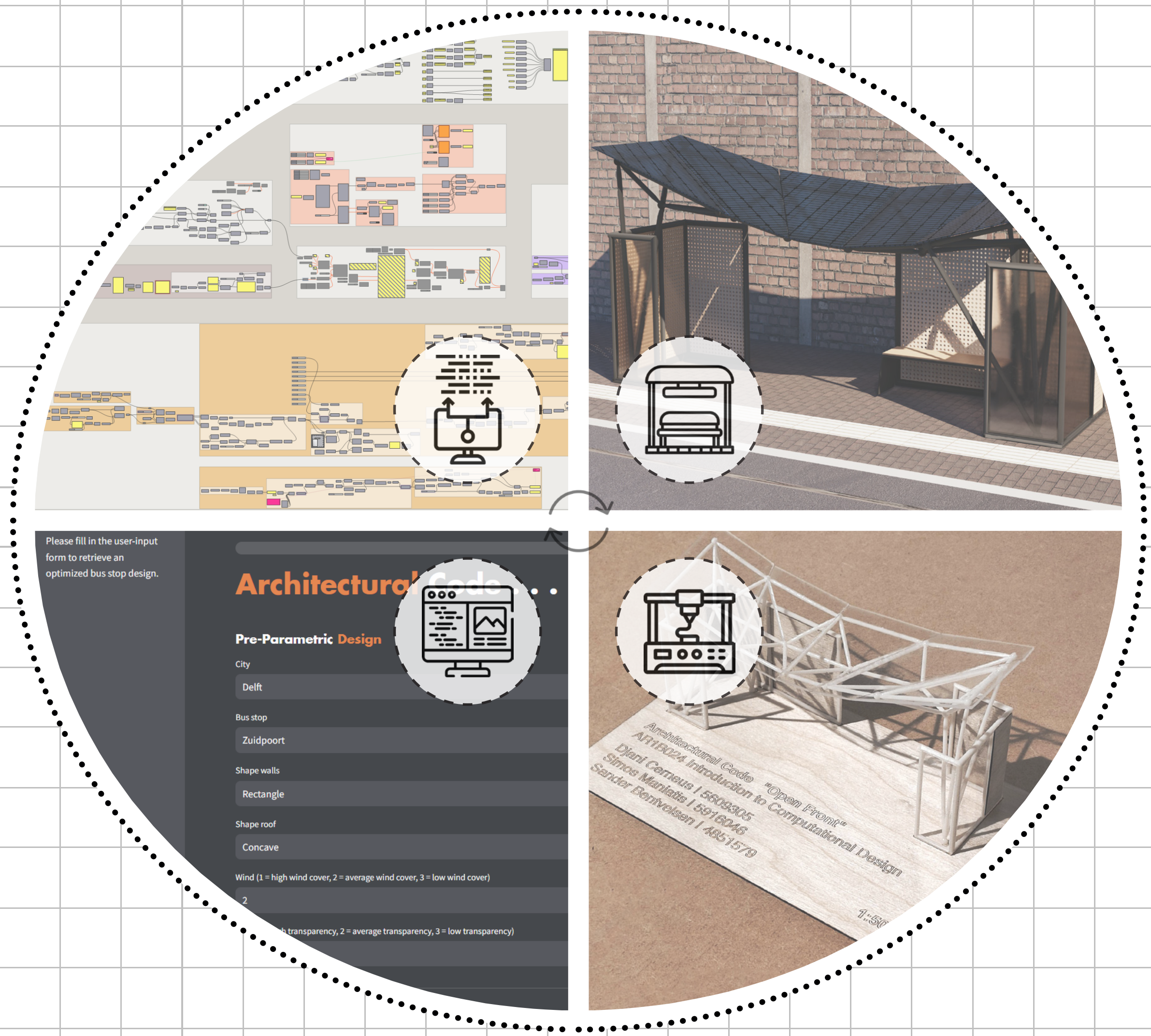
ARCHITECTURALCODE

PARAMETRIC BUS STOP DESIGN



MOVIE

https://youtu.be/Oy3-ZEjXaAc

"Where the art of architectural design meets the power of computational design."

Djani Cerneus | 5609305
Simos Maniatis | 5916046
Sander Bentvelsen | 4851579

AR1B024 Introduction to Computational Design | Dr. S. Asut

02-02-24

CONTENTS

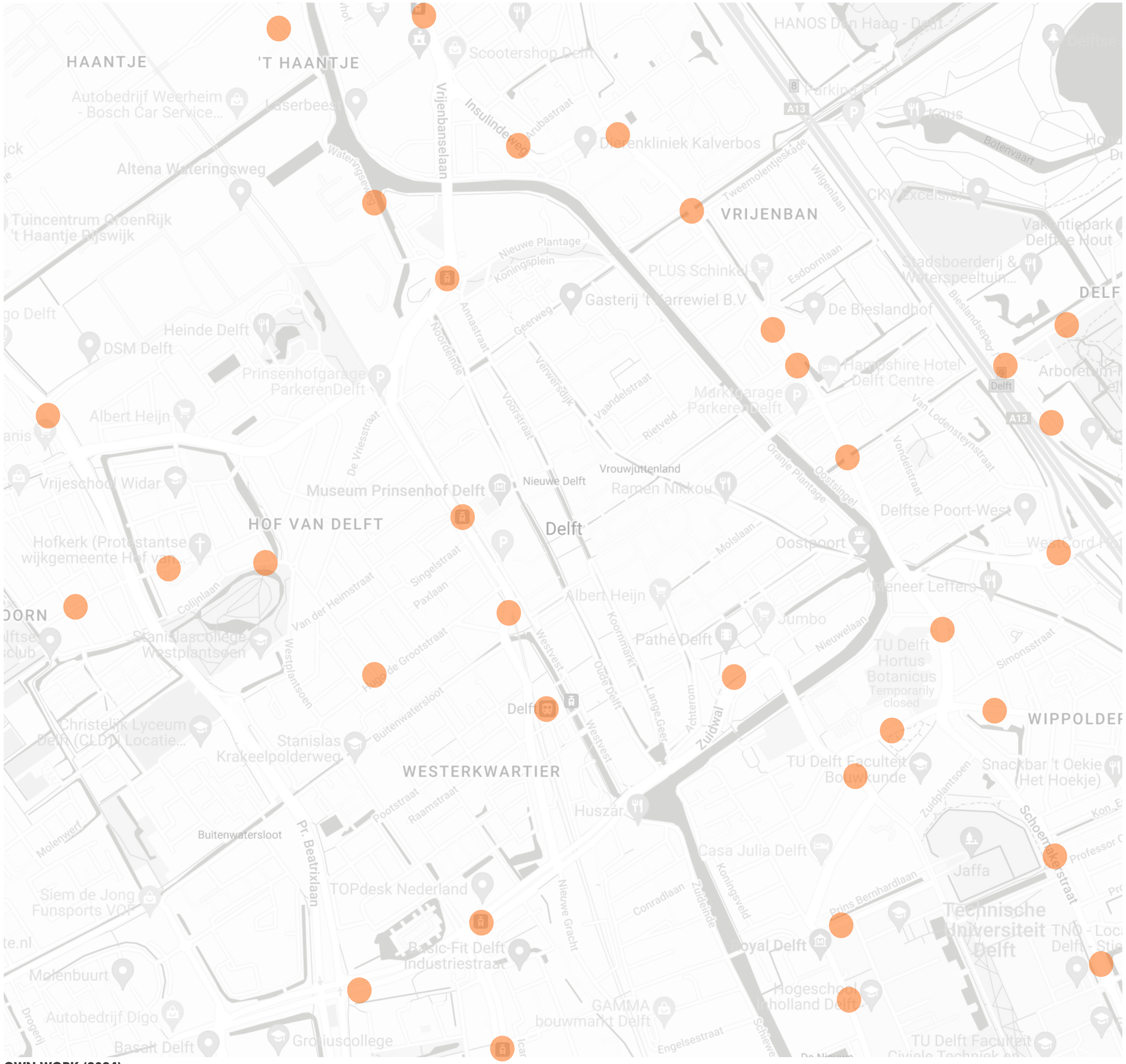
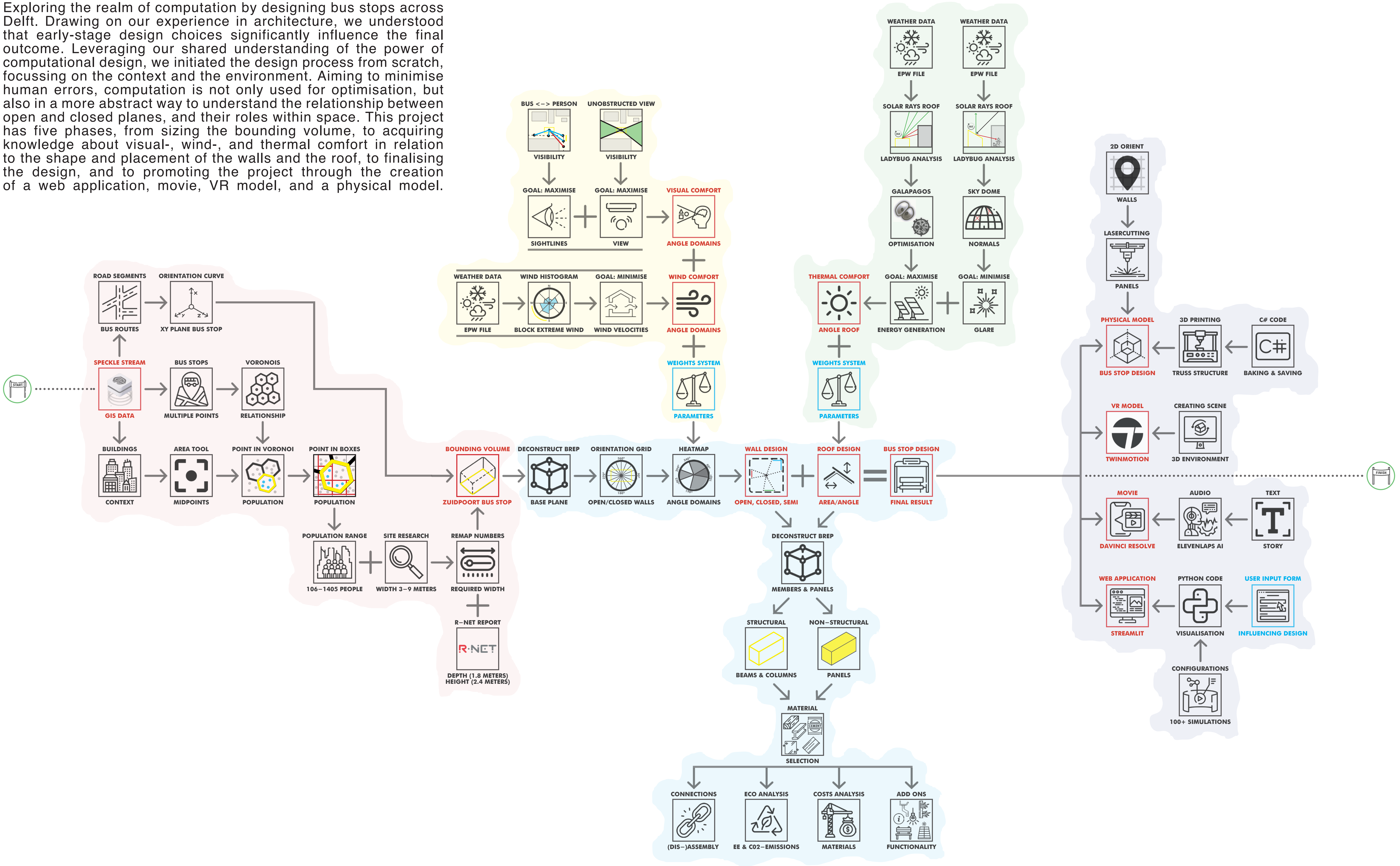


FIGURE 02 BUS STOP LOCATIONS ACROSS DELFT
OWN WORK (2024)

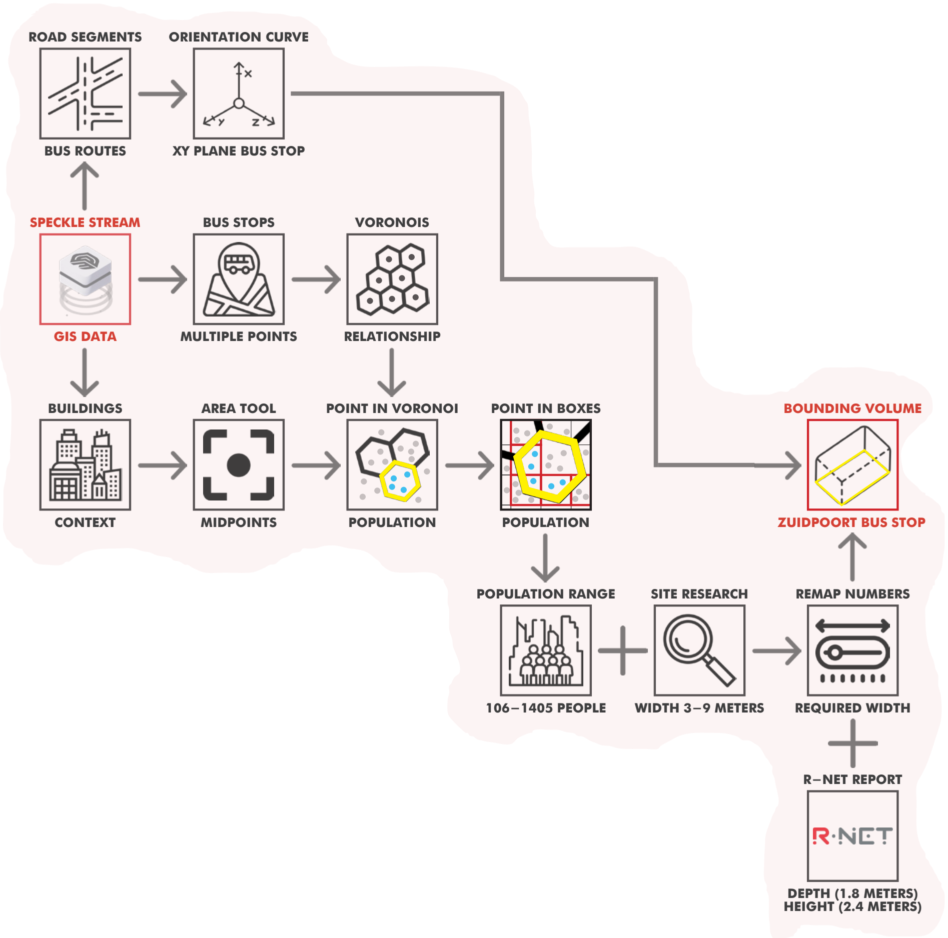
INTRODUCTION

Exploring the realm of computation by designing bus stops across Delft. Drawing on our experience in architecture, we understood that early-stage design choices significantly influence the final outcome. Leveraging our shared understanding of the power of computational design, we initiated the design process from scratch, focussing on the context and the environment. Aiming to minimise human errors, computation is not only used for optimisation, but also in a more abstract way to understand the relationship between open and closed planes, and their roles within space. This project has five phases, from sizing the bounding volume, to acquiring knowledge about visual-, wind-, and thermal comfort in relation to the shape and placement of the walls and the roof, to finalising the design, and to promoting the project through the creation of a web application, movie, VR model, and a physical model.



01 GIS DATA

This chapter delves into the use of GIS data and its integration into the design process of bus stops across Delft. It emphasizes the methods employed to calculate the population around each bus stop, ultimately determining the dimensions of their bounding volume.



1.1 POPULATION

Aiming to attain an accurate visual representation of the context of Delft, the Grasshopper algorithm restructures the Speckle data provided for the buildings and their elevation, the bus routes, and the bus stops. In addition to GIS data, population size data is also given per square kilometer, categorised into five age groups ranging from 0 to 75 years old. The initial step in the population calculation was to get grip of the relationship between the various bus stops distributed across Delft. To illustrate this relationship effectively, the Voronoi method was applied, delineating the border around each bus stop and showcasing the area for which each bus stop is responsible (see Figure 03). This method involves taking each bus stop location as a point and connecting lines between each point. At the midpoint of each line, a 90-degree line is drawn.

FIGURE 03 VORONOI METHOD

OWN WORK (2024)

These Voronoids served as the foundation for two calculations. The first calculation involved obtaining all the midpoints of the buildings and filtering out those that were not within the Voronoi. The second calculation was more intricate, given that the population data was provided per square kilometer. This led to situations near the border of each Voronoi where not all buildings inside each square needed to be considered in the population calculation, as they were outside the Voronoi (see Figure 04). To address this, the area of the buildings inside both the square and the Voronoi was calculated and then divided by the total area in each square. This yielded a population reduction factor for each square, with inner squares experiencing no reduction as 100% of the buildings were inside both the Voronoi and the square. Employing a looping component named Ademone, this process was iterated over 150 times to calculate for each Voronoi in Delft. These calculations allowed for a precise determination of the population range (106 to 1405) across bus stops in Delft.

1.2 BOUNDING VOLUME

Based on a technical report by RNET, the company responsible for most bus stops across the country, it can be inferred that an ideal bus stop should have dimensions of approximately 1.8 meters in depth and 2.4 meters in height. The width, however, is dependent on the anticipated number of users for the bus stop (RNET, 2013). This is where the calculated population range proves valuable. Complemented by on-site research, which involved measuring the width of each bus stop in Delft and subsequently calculating the range of widths, a domain was established for the width of the bounding volume, ranging from 3 to 9 meters, contingent upon the size of the population around it (see Figure 05). These bounding volumes serve as the foundation for the design process, as they represent the space in which the relationship between open and closed planes will be investigated.

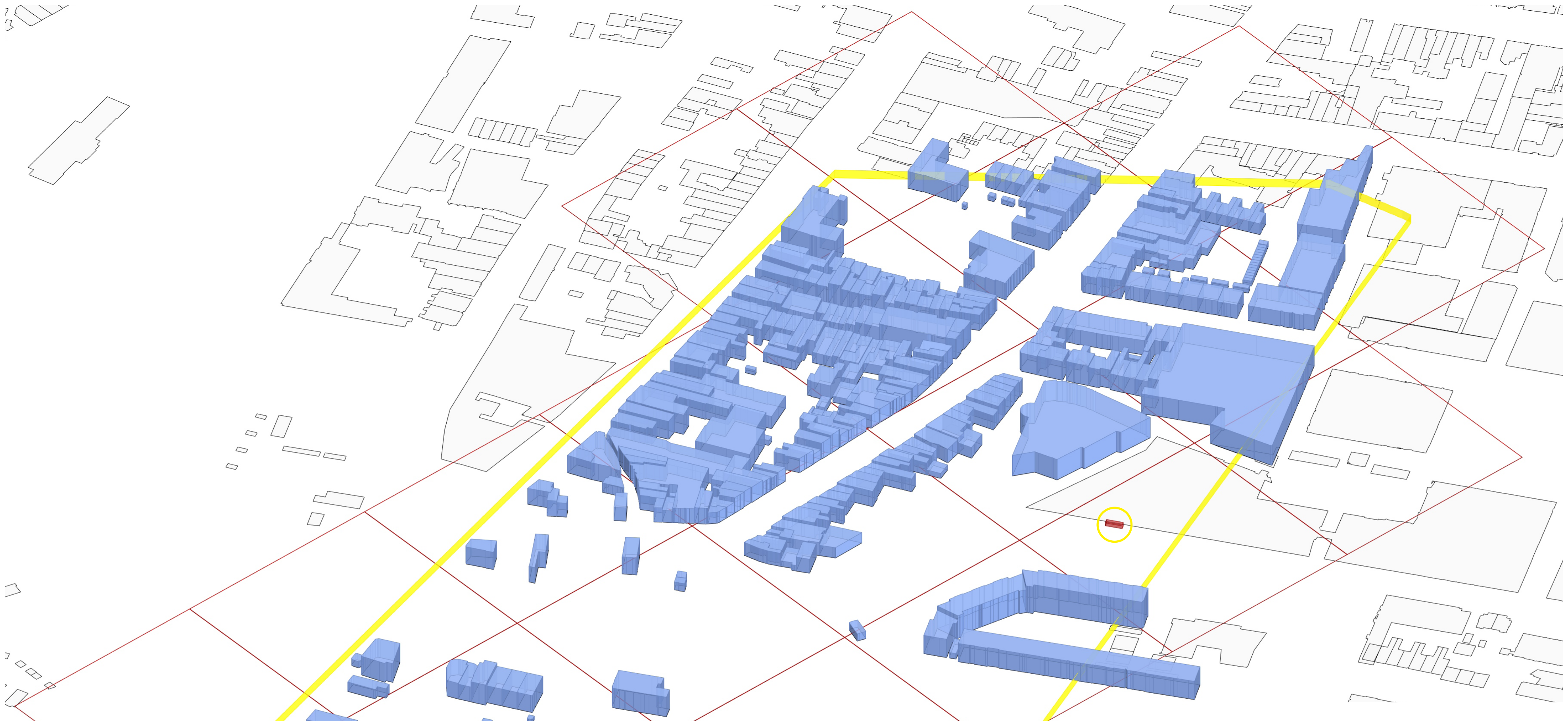


FIGURE 04 POPULATION CALCULATION

OWN WORK (2024)

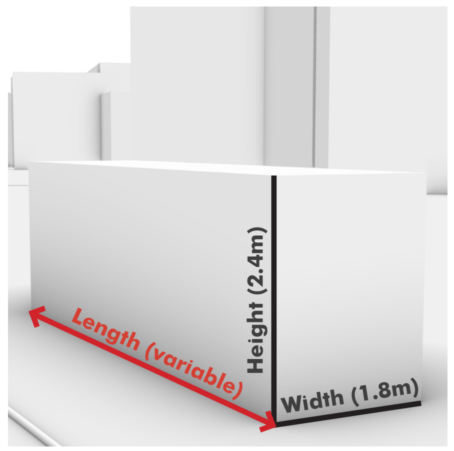
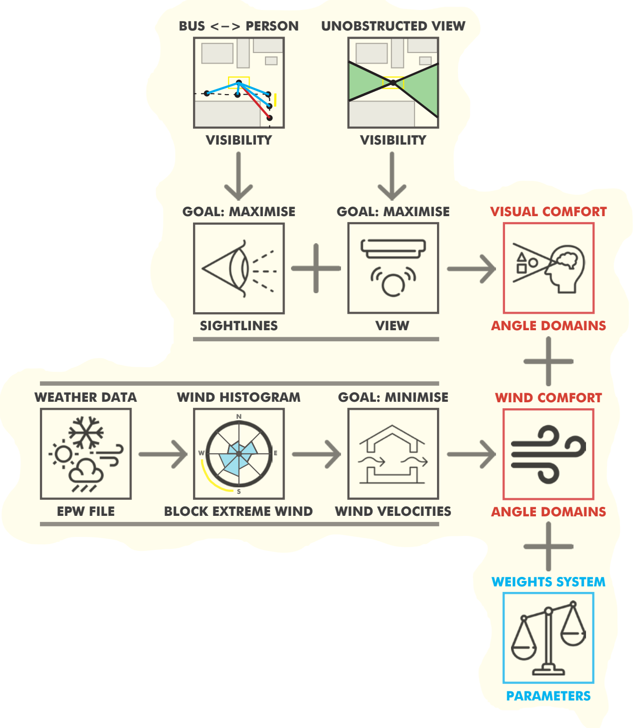


FIGURE 05 BOUNDING VOLUME

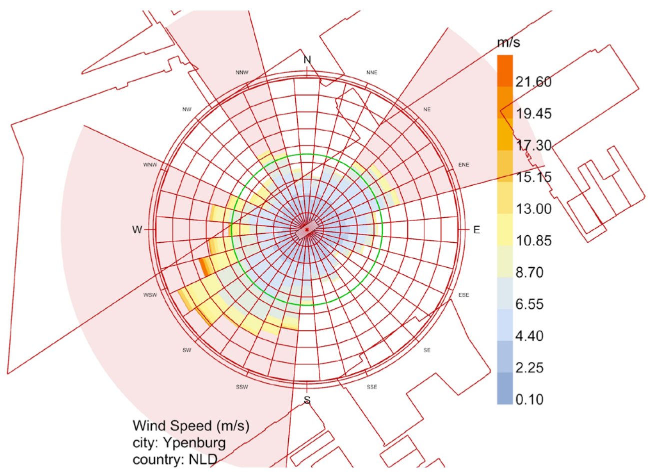
OWN WORK (2024)

This chapter delves into the optimisation of the bus stop design, focusing on its walls in particular. Computation is used in an abstract manner to understand the relationship between open and closed planes and their roles within bounding volumes. Concerning the walls, this relationship is investigated by determining optimal angles for the placement or absence of walls. This exploration includes a wind analysis providing positive angles and two visual analyses yielding negative angles, indicating areas where walls may be undesirable or where transparent walls are preferred.



2.1 WIND

To determine the optimal placement of solid walls, wind data is imported from the EPW file of Ypenburg (see Figure 07). The objective is to position walls in areas where the wind predominantly originates. By summing these wind speeds, a list can be generated, representing the total wind units throughout the year per 5 degrees. A custom Python script, along with a number slider to set the maximum wind units, is employed to create domains. These domains can later be translated to identify the appropriate locations for wall placement.



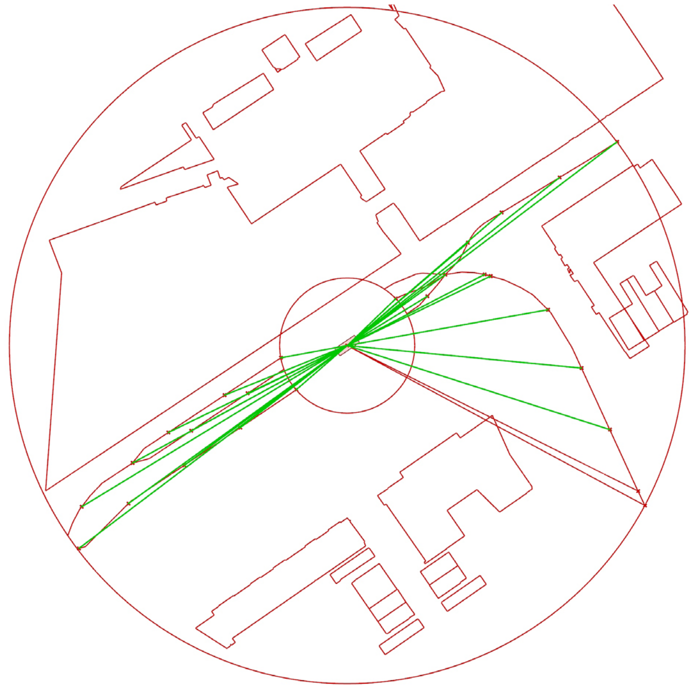
WIND ANALYSIS

FIGURE 06

OWN WORK (2024)

2.2 SIGHTLINES

To ensure visibility between the bus driver and the people at the bus stop, as well as vice versa, a sightline analysis is conducted. Roads within 20 to 100 meters of the bus stop are identified, and rays are evenly distributed from these roads to the bus stop. These rays are then checked for intersections with buildings, as these obstructed roads would not provide visibility either way. The sightlines that successfully reach the bus stop are subsequently utilised to ascertain which parts of the walls need to be transparent.



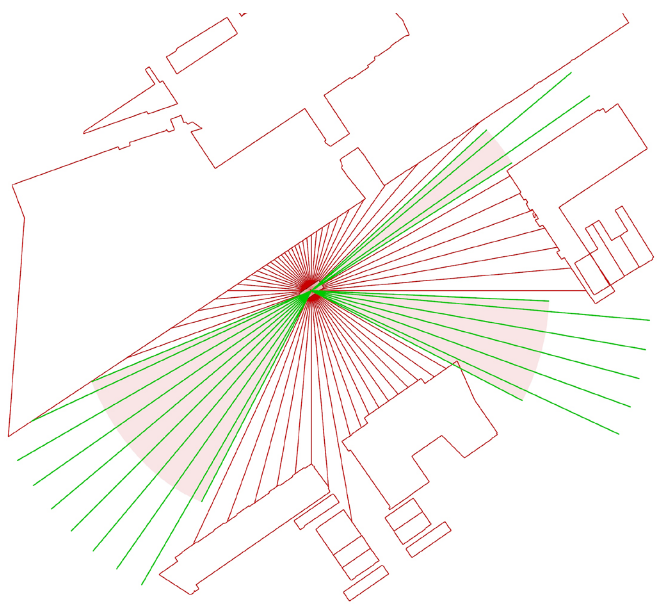
SIGHTLINES ANALYSIS

FIGURE 07

OWN WORK (2024)

2.3 VIEW

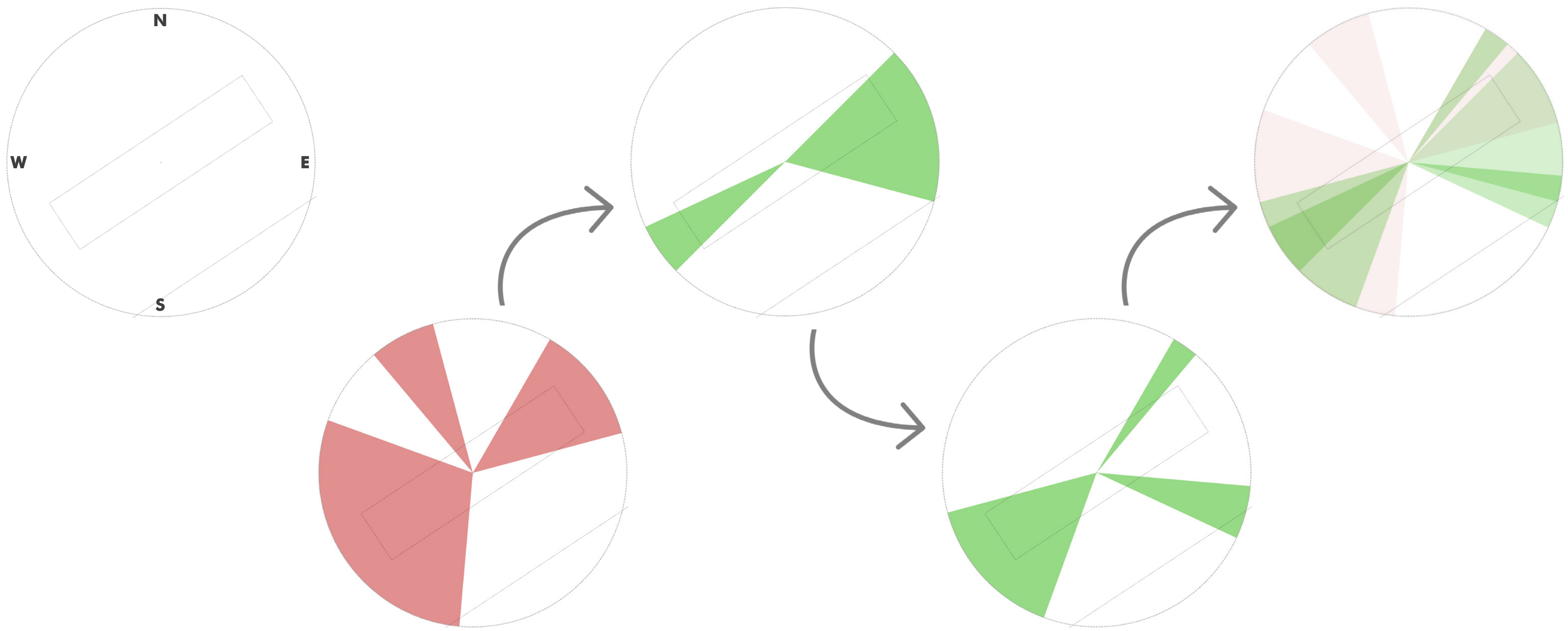
When people wait at a bus stop, it's preferable for them to have a pleasant view of their surroundings, which is often achieved best with wide-open areas in front of them. To ensure these pleasing views, another ray-casting approach is used to assess the distance of buildings within specific angles. By establishing a minimum distance, angles likely to provide a pleasant view are identified. These angle domains are then considered when deciding which parts of the walls should be made transparent.



VIEW ANALYSIS

FIGURE 08

OWN WORK (2024)



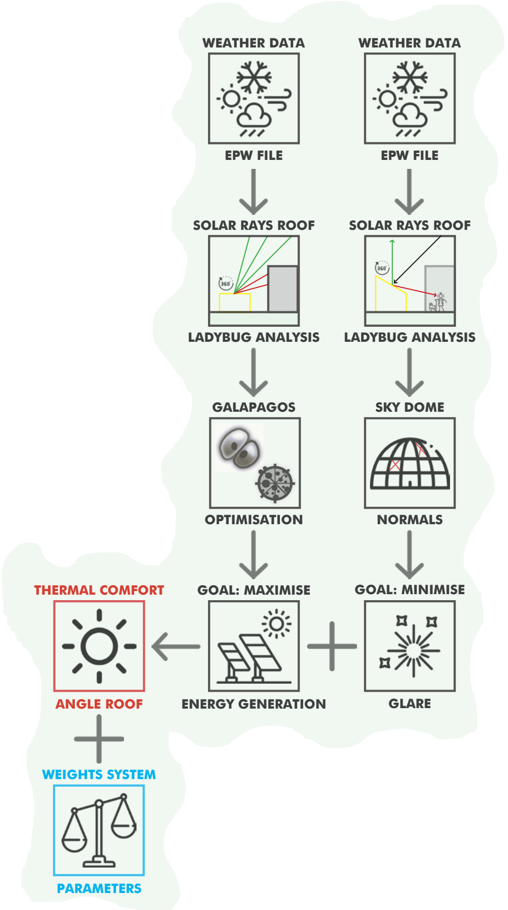
ANGLE DOMAINS

FIGURE 09

OWN WORK (2024)

03 ROOF OPTIMISATION

This chapter continues to delve into the optimisation of the bus stop design, this time focusing on its roof. Following the same philosophy as the wall placement, normal angles of the roof are calculated prior to the actual design process. These normals depend on two analyses: one maximising energy generation, resulting in positive angles, and the other minimising glare, yielding negative angles.



ENERGY GENERATION ANALYSIS

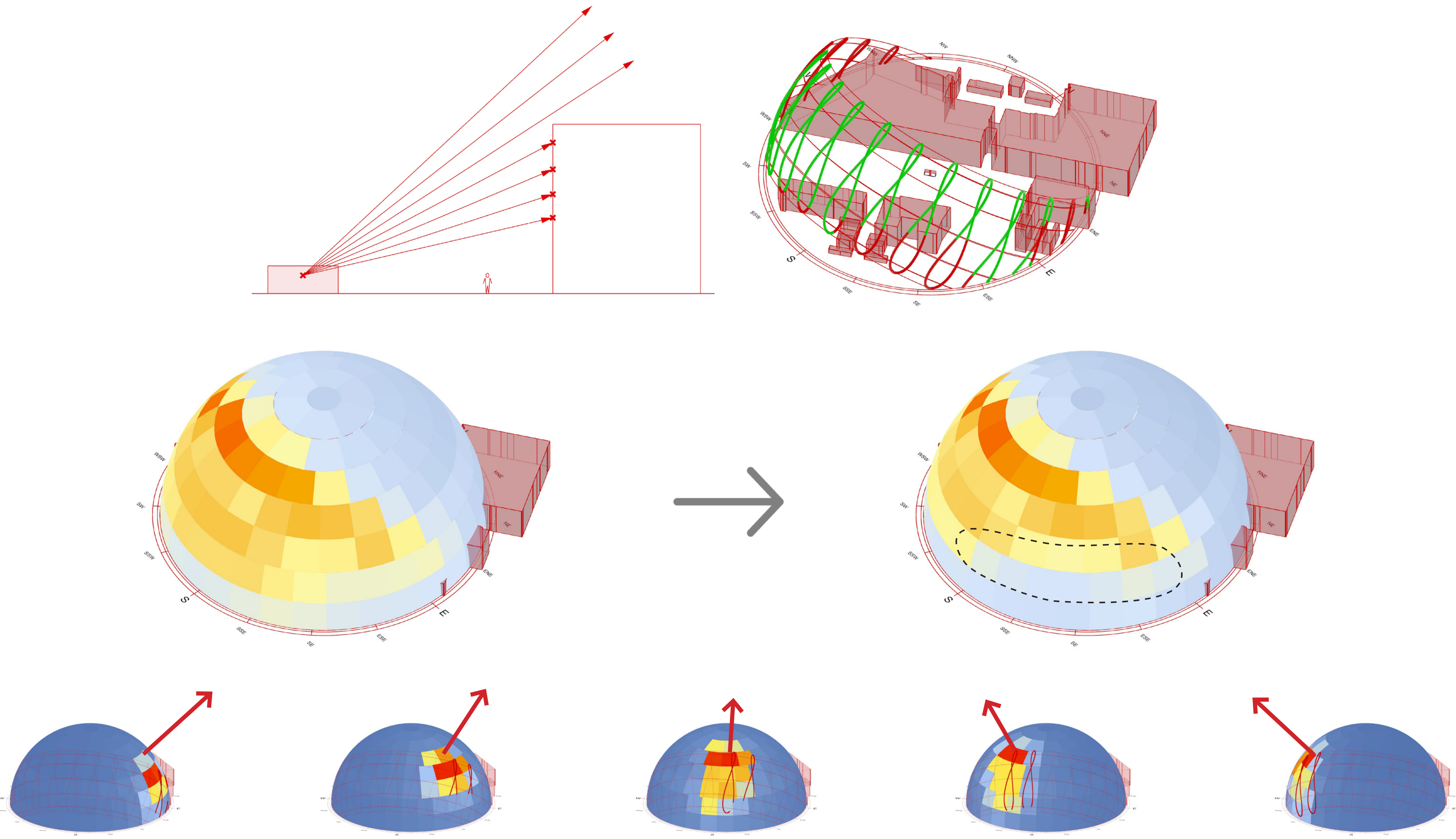


FIGURE 10

OWN WORK (2024)

3.1 ENERGY GENERATION

Solar panels are an effective way of generating energy throughout the day. However, a challenge with solar energy is its tendency to peak when the sun is high and sharply decline as the sun sets. Tilted solar panels can mitigate this issue by aligning with the sun's direction throughout the year, allowing for a more extended and effective energy generation period.

To discretise the final elements in the design, the Grasshopper script divides the day into five parts: 10:00-12:00, 12:00-14:00, 14:00-16:00, 16:00-18:00, and 18:00-20:00. These periods are chosen as they correspond to times when the sun is above the horizon and mostly unobstructed by surrounding buildings.

To account only for effective solar hours, a ray-casting step is performed to eliminate the hours of the year (HOYs) when the sun is blocked by surrounding buildings (see Figure 10). The remaining HOYs are then subdivided into five time groups, and the script calculates the radiation domes for these time periods throughout the entire year. Averaging these values over the skydome enables the determination of normal angles that optimise energy generation for specific time groups at this particular location.

3.2 GLARE

While generating energy with a solar roof is beneficial, it should be executed in a way that ensures safety and comfort for its surroundings. A glare analysis is conducted to prevent road users from being blinded by the reflected sunlight. This is achieved by collecting unobstructed road vectors cast from eye height to the roof of the bus stop, similar to the sightlines analysis but in 3D.

By normalising these vectors and then combining them with unobstructed solar vectors, a new vector is formed, representing the normal direction of the surface that may cause glare at that location.

Optimising the parameters to cast around 10 million rays on oversampled skydome meshes results in a shadowdome that illustrates the normal directions causing significant glare throughout the year.

Following that, the optimal roof angles are rotated towards the zenith to find an angle that reduces both the height of the roof and minimises potential glare complications. If this is not feasible (the normal angle causes excessive glare over the entire range of rotation to the zenith), however, it is simply culled, and the roof is discretised into four separate pieces instead of five.

GLARE ANALYSIS

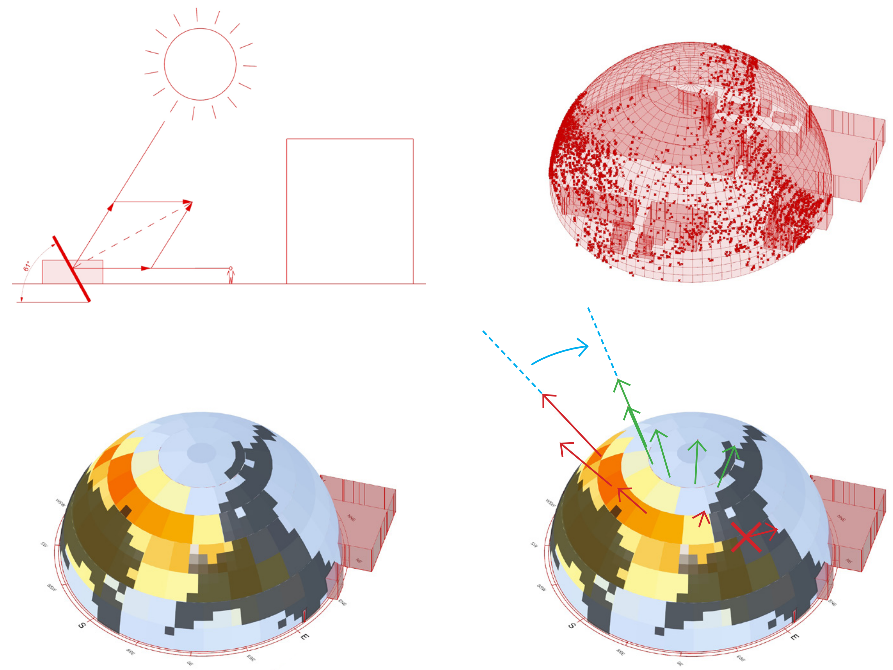
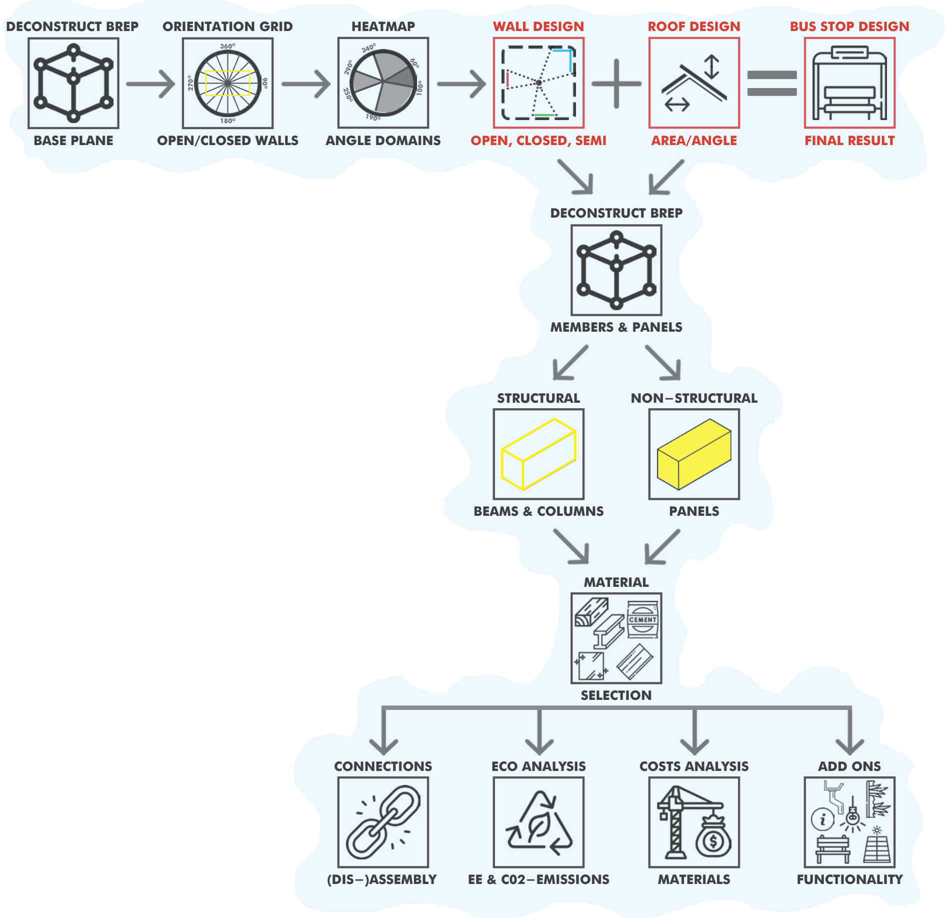


FIGURE 11

OWN WORK (2024)

This chapter delves into the final design process of bus stops across Delft, translating the calculated optimum wall and roof angles into tangible geometries. The process can be divided into two main phases: one emphasizing the computational methods used to generate the bus stop geometries, and the other discussing materialisation, environmental impact, costs, and additional functionalities.



4.1 WALL DESIGN

Starting with walls, optimal angles for closed and transparent walls overlay on the chosen bus stop shape. An additional void is inserted at the front for easy entry and exit. The initial step involves creating curves indicating wall placements, ensuring a minimum size. These curves are then split at transparent parts (green). Subsequently, all curves are checked for length and possibly split further based on panel size limitations. Once these curves are generated, the rest of the geometry is created using predefined parameters like bus stop height, panel width, offset, and structural pole width (see Figure 12).

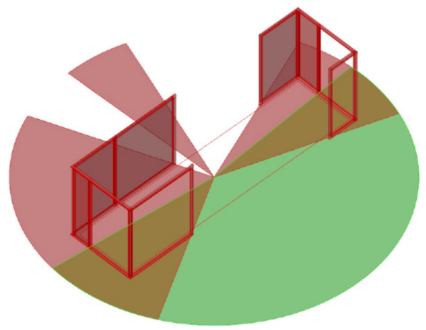
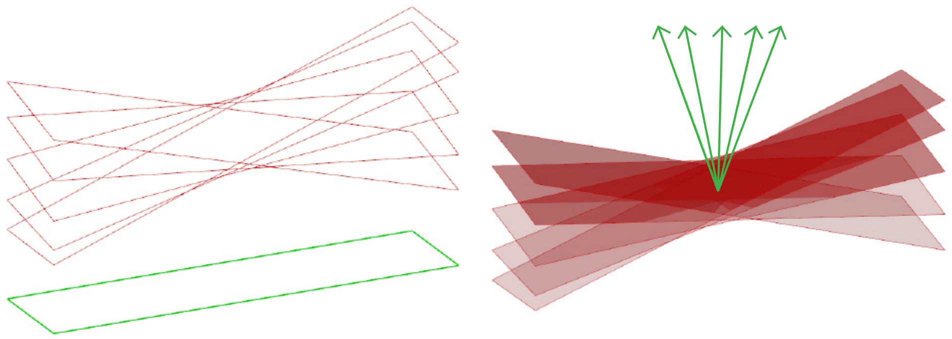


FIGURE 12
OWN WORK (2024)

4.2 ROOF DESIGN

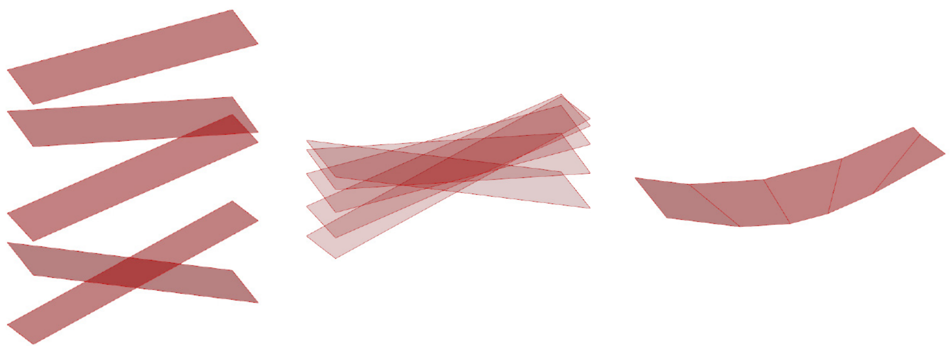
The generation of the roof is a bit more complex and requires an iterative optimising algorithm. First the surfaces are cut out as projected by the required shape, again this can be any shape. The normals of these surfaces correspond to those calculated in the prior steps.



NORMALS TO SURFACES

FIGURE 13
OWN WORK (2024)

The next step is to move the surfaces up and down and use them as intersecting planes, only keeping the top surfaces as they are the ones that remain. An optimiser iteratively moves the surfaces and is rewarded more if the areas of the remaining surfaces are relatively equal and the shortest edges of the surfaces are longer than 800mm. This is necessary for the generation of the space frame.

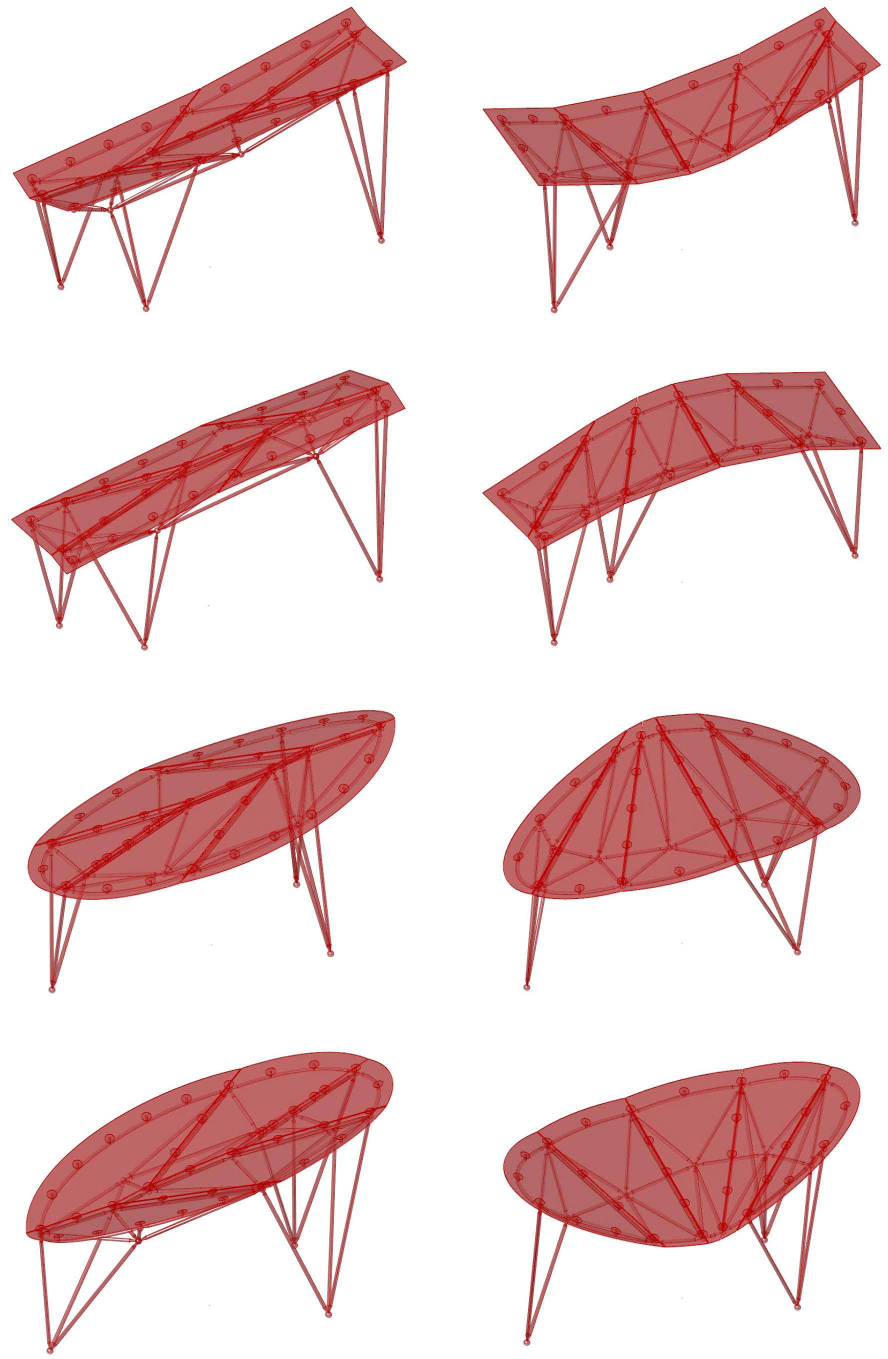


SURFACE PLACEMENT

FIGURE 14
OWN WORK (2024)

When the final height of the surfaces is determined it's time to generate the geometry. First the outside curves are offset and moved down, so the space frame will generate underneath the panels. Using some intricate listing, a network of curves is generated that creates a stiff frame underneath. Then the columns are added by finding the three closest points on the zy plane, to keep them stable and as vertical as possible. Next the truss is generated and can be customised based on 11 parameters. Giving full flexibility and possibility to further optimise if required. Finally the panels are generated and slightly offset as to make them possible to manufacture to required tolerances.

NORTH–SOUTH　EAST–WEST

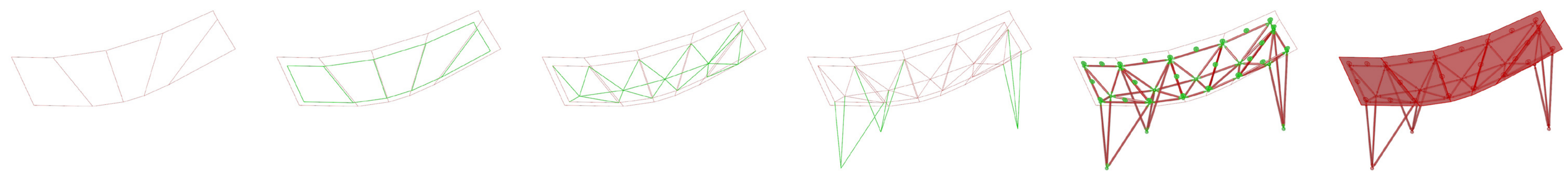


POTENTIAL CONFIGURATIONS

FIGURE 16
OWN WORK (2024)

As a proof of concept here are some various roof shapes the optimiser is capable of generating. The north-south facing bus stops mimicking (perhaps not so coincidentally) nature for optimal equal solar performance throughout the day.

WALL PLACEMENT



VOLUME CEATION

FIGURE 15
OWN WORK (2024)

4.3 CONSTRUCTION

With the finalisation of the bus stop geometry, discussions now focus on materialisation, environmental impact, costs, and additional functionalities. The process begins by selecting the appropriate construction material, a choice between aluminium and steel. Due to the similar (space truss) nature of each bus stop (see Figure 17), Circular Hollow Sections (CHS) are chosen for both functional and aesthetic reasons. Structural calculations for one of the main columns, representing a cantilevering beam, led to the selection of CHS-60.3/4 for aluminium and CHS-42.4/4 for steel, where the first number signifies the diameter and the second number the wall thickness. This difference reflects the varying strength of the two materials. Subsequently, a Grasshopper script conducts environmental and cost analyses based solely on the mass of the structural material required, excluding walls and roof from the calculation. The analysis concludes that steel is a more suitable construction material, as choosing aluminium would result in higher embodied energy, carbon emissions, and costs (see Figure 18).

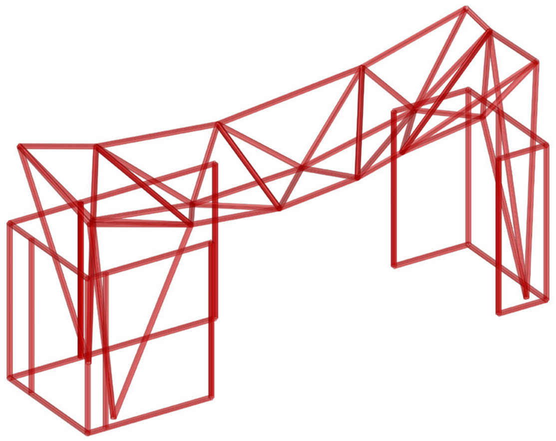
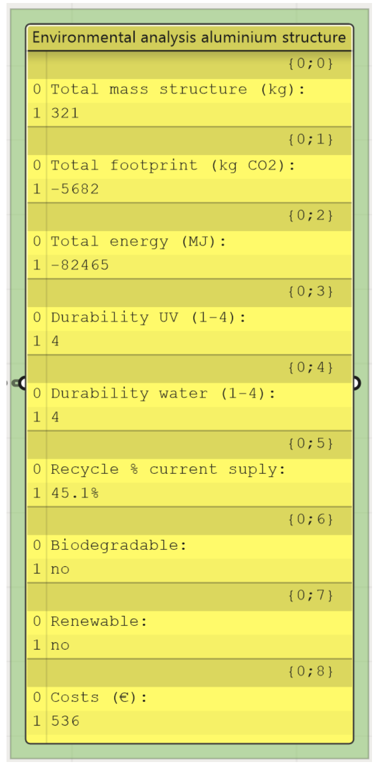


FIGURE 17 CONSTRUCTION

OWN WORK (2024)



VS

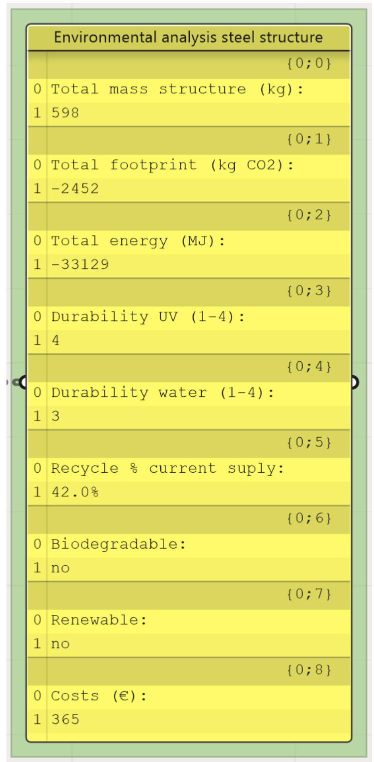


FIGURE 18 ENVIRONMENTAL AND COST ANALYSIS

OWN WORK (2024)

4.4 VISUALISATION

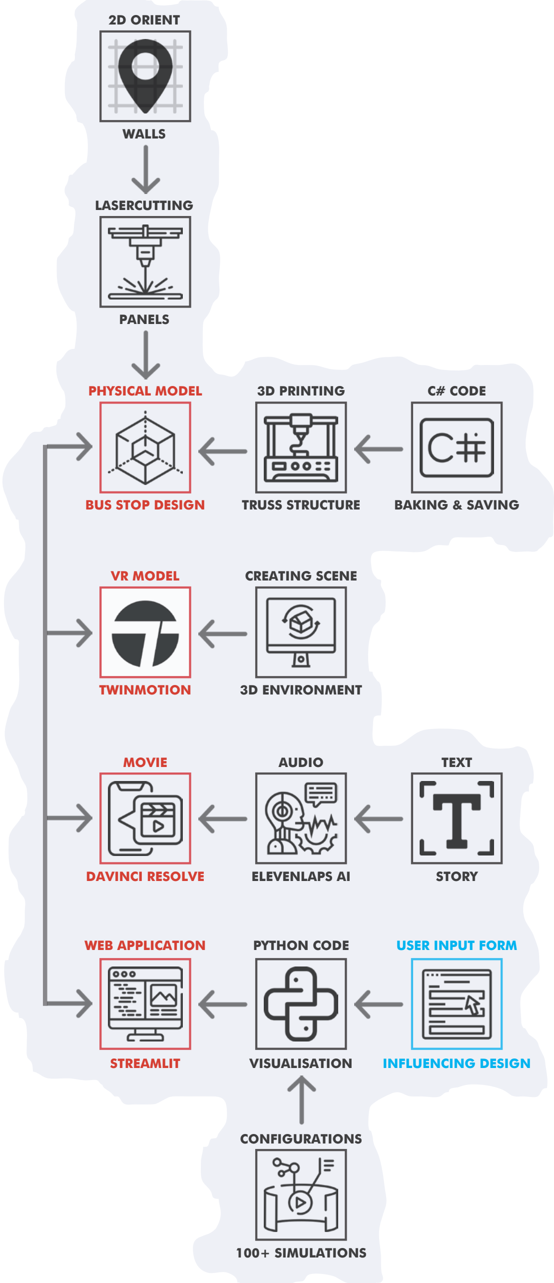
To finalise the bus stop design and to give it some character, the materialisation of the closed and transparent walls takes place, using perforated metal panels and glass, respectively (see Figure 19). Options are abundant due to the structural system's versatile nature, including choices like wooden panels, vegetation panels, composite panels, etc. As previously mentioned, PV panels are installed on the roof to generate energy year-round. The seams between each cell are meticulously sealed to maintain water coverage. Additionally, a gutter can be placed along the roof outline for better control of water management. To enhance functionality, lighting can be mounted at the bottom of the roof truss structure, ensuring visibility, comfort, and safety. This lighting can be connected to the output of the PV panels to create a self-sustaining system. Lastly, an information sign and a bench can be included.

FIGURE 19 VISUALISATION FINAL BUS STOP DESIGN

OWN WORK (2024)

05 EXPORTS

This final chapter delves into the process behind creating a web application made in Python, accompanied by a movie, as well as a VR model, and a physical model, all aimed at promoting the project.



5.1 WEB APPLICATION

With Architectural Code, we don't just offer the adaptability of using our Grasshopper algorithm on different bus stop locations, adjusting to site-specific data and generating an optimised bus stop geometry each time. We aim to do more. The challenge is to give our users the opportunity to design bus stops across Delft according to their specific desires, both functionally and aesthetically. To achieve this, we created a web application built in Python using Streamlit.

Upon entering the web application, users encounter a user-input form with various dropdown options on different aspects of the bus stop design (see Figure 20). Initially, three bus stop locations in Delft can be selected: Zuidpoort, Hugo van Rijkenlaan, and Christiaan Huygensweg (see Figure 21). Each location presenting its own unique challenges and opportunities influencing design choices. While navigating the form, users can select the structure's shape, being a rectangle or an ellipse. Also the shape of the roof can be made concave or convex, understanding its impact on rain management. Finally, the importance of wind cover and the level of transparency, on a scale from 1 to 3, can be specified. Once the user is fully satisfied and presses the submit button, a dynamic 3D model viewer (Pyvista) opens a customised bus stop design, adapting instantly to their input. Behind the scenes, over 100 simulations have been executed in Grasshopper, each saved with a unique code. When the user-input form is submitted, this code finds its match among the simulations, bringing the user's vision to life. The primary message this web application aims to convey is the power of computational design, emphasizing its remarkable adaptability.

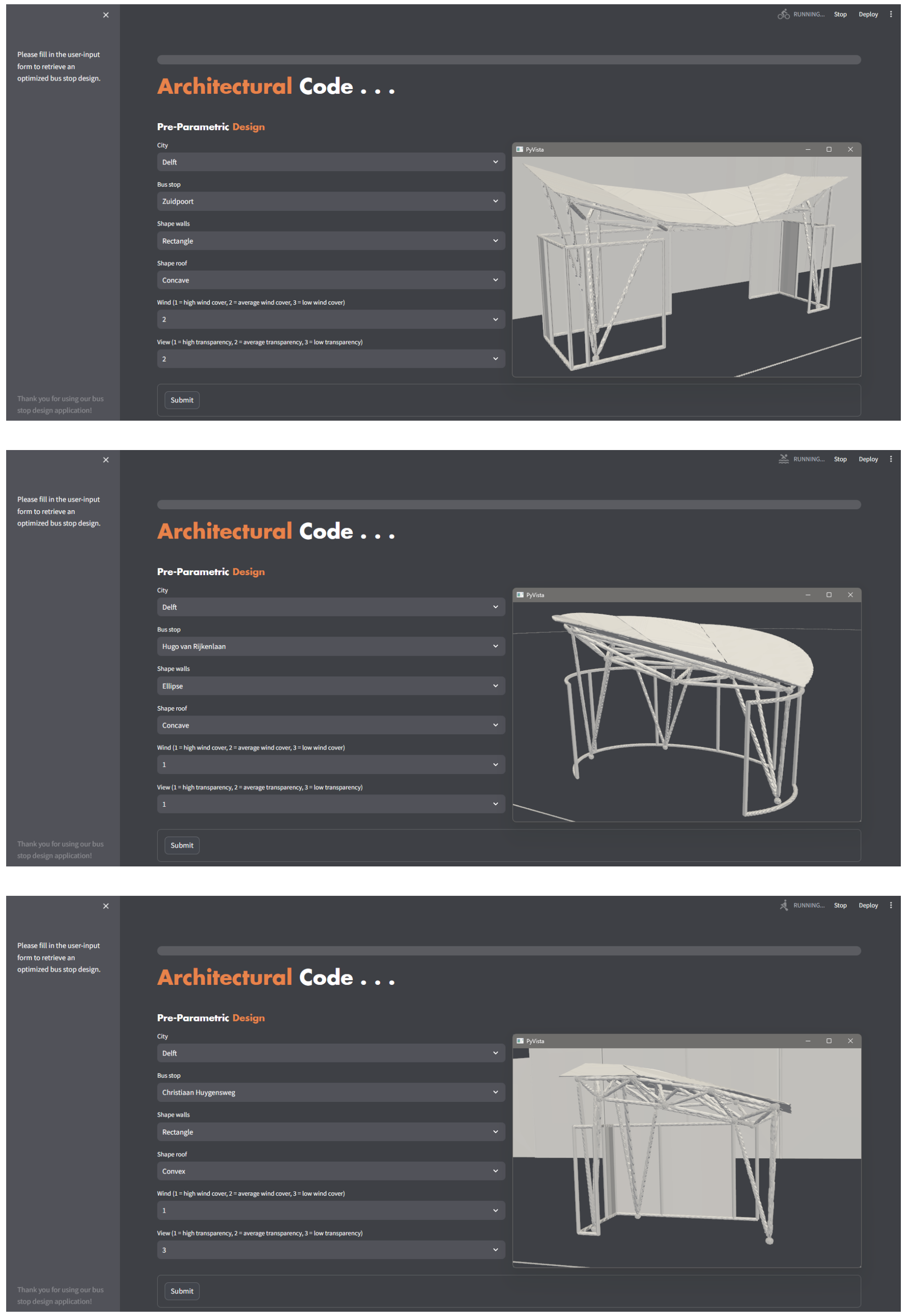


FIGURE 20 CHANGING PARAMETERS AFFECTING BUS STOP DESIGN

OWN WORK (2024)

FIGURE 21 PROJECT LOCATIONS

OWN WORK (2024)

5.2 MOVIE

As aforementioned, a supporting movie was made to go along with the web application, showing off its features and offering instructions on how to operate it. ElevenLaps AI was used to create the voice-over for the video after the content was created. Basic screen-recording software was used to record the video. The audio and video data were then imported into DaVinci Resolve (see Figure 22), where the movie was put together.

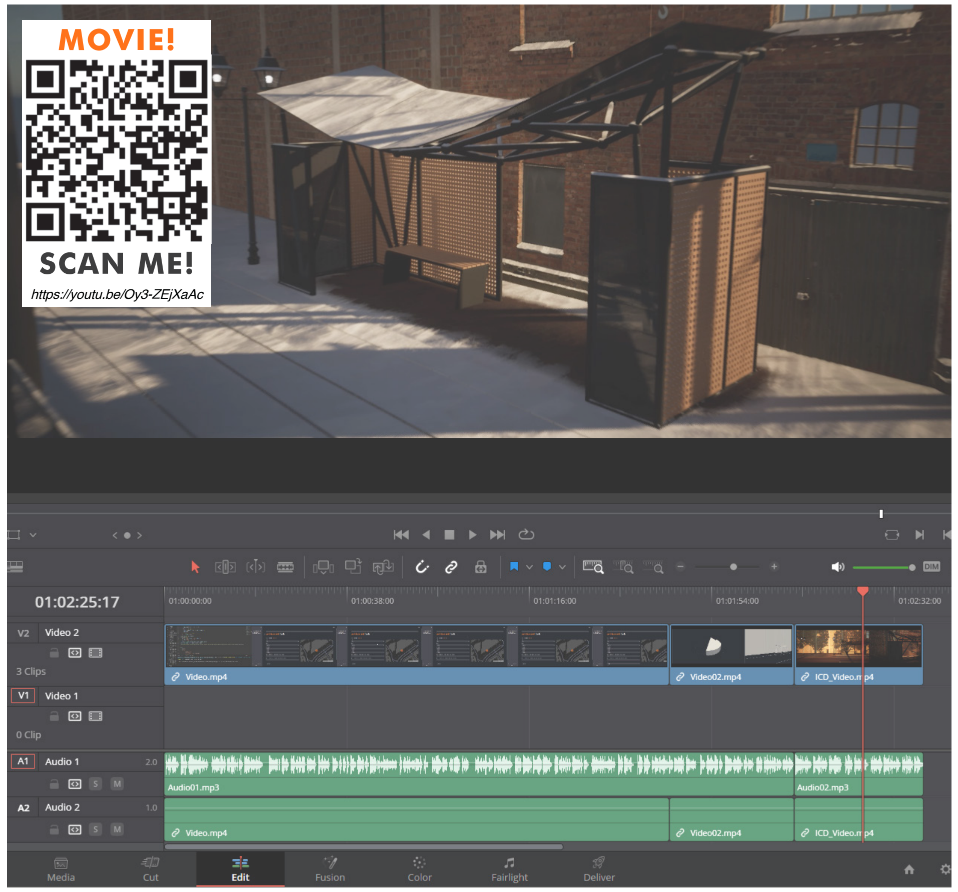


FIGURE 22 DAVINCI RESOLVE MOVIE EDITOR INTERFACE

OWN WORK (2024)

05 EXPORTS

5.3 VR MODEL

To promote the project more effectively, a VR model of a random configuration of the bus stop design, at the Zuidpoort bus stop, was created. The challenge was to represent the urban context as accurate as possible, especially since the Speckle GIS data lacked detailed 3D geometry. To address this, Google Maps was used to gather a comprehensive understanding of the context.

Starting with importing the basic GIS data into Rhinoceros: the foundational elements of the context, such as topography and key infrastructural features, were laid out (see Figure 23). Since the GIS data did not provide any detailed 3D geometries, Rhinoceros was used to add the essential urban buildings and landscaping. The objective here was not to create a 1-to-1 replica of the context, but rather to model an environment that was similar enough to provide a realistic scene for the VR model of the bus stop.

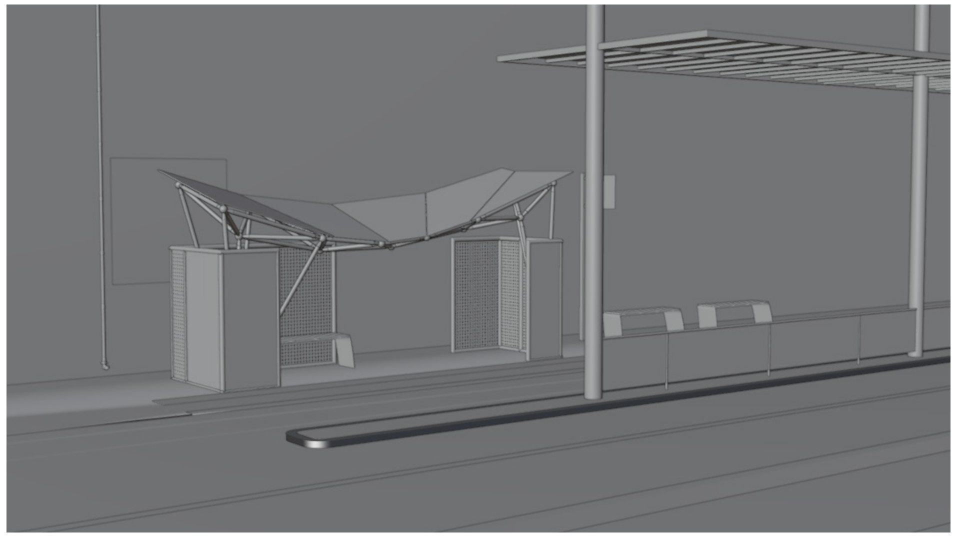


FIGURE 23 IMPORTING BASIC GIS DATA IN RHINOCEROS

OWN WORK (2024)

After establishing the urban layout in Rhinoceros, the model was transferred to Blender where the focus was on enhancing the model with detailed textures, architectural elements, and realistic lighting.

The model was then exported to Twinmotion (see Figure 24). In this phase, the environmental aspects like weather conditions and lighting were set up to match the envisioned scenario, ensuring the bus stop design was presented within a believable urban setting.

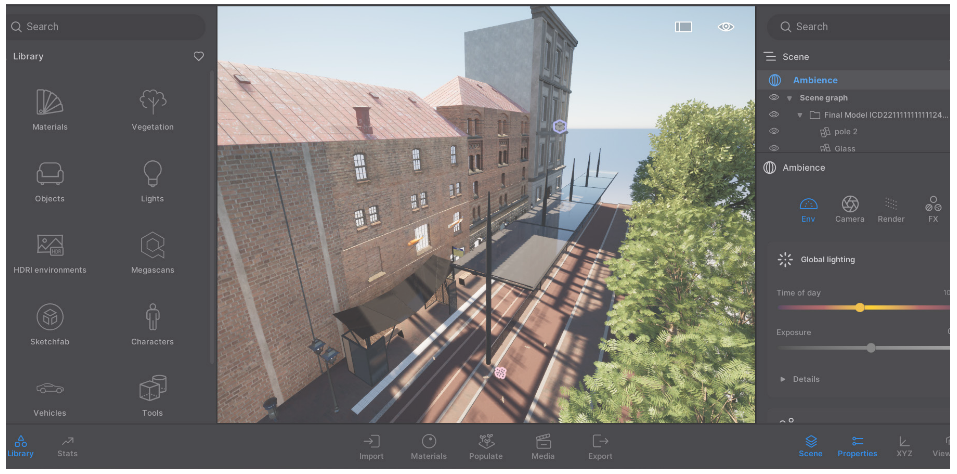


FIGURE 24 TWINMOTION EDITING

OWN WORK (2024)

For the final touch, the camera within Twinmotion was carefully set. This step was critical for showcasing the bus stop effectively, adjusting camera angles, lens settings, and compositions to highlight the design. The aim was to present the bus stop as a natural part of the context, set within a realistically detailed, yet not exact, replication of its real-world surroundings. This approach allowed for creating a compelling and immersive VR model that accurately conveyed the essence and context of the bus stop design.

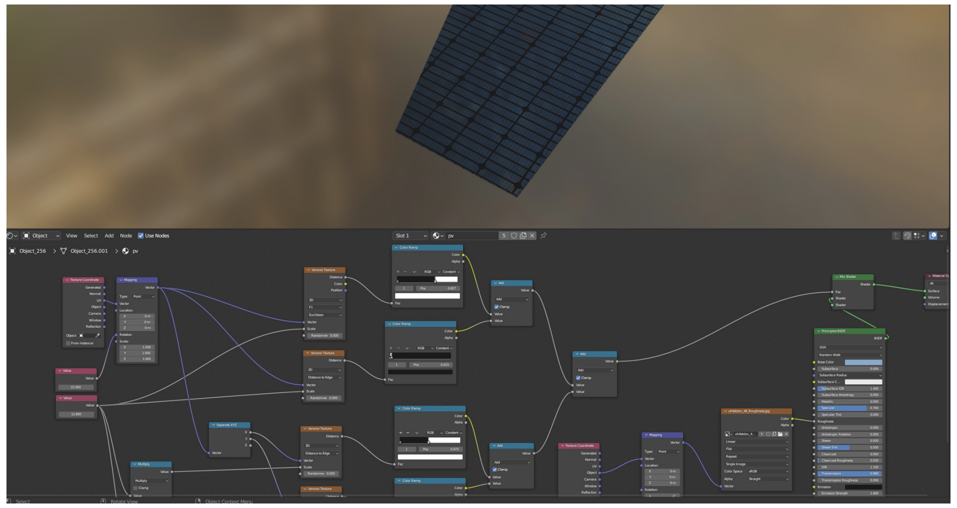


FIGURE 25 OPTIMISING VR SETTINGS

OWN WORK (2024)

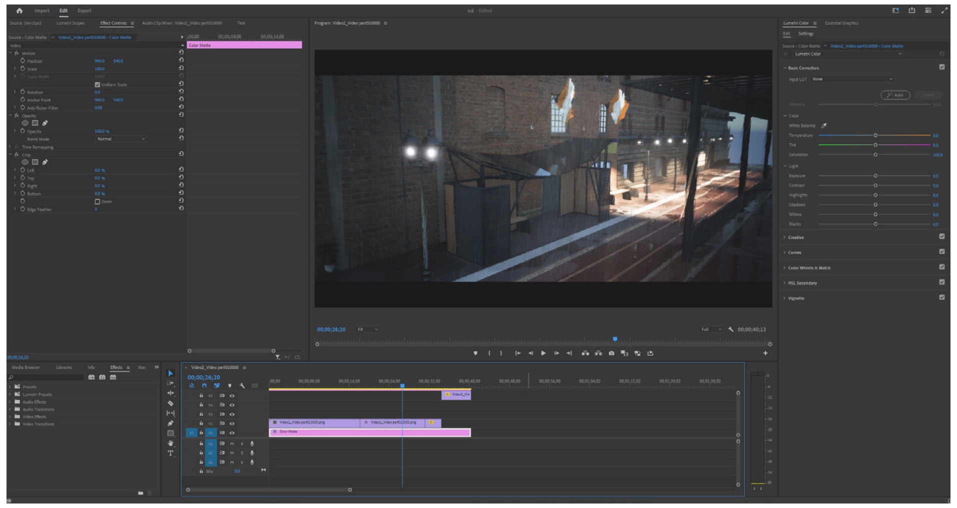


FIGURE 26 CREATING MOVIE

OWN WORK (2024)

FIGURE 27 FINAL VR ENVIRONMENT

OWN WORK (2024)

5.4 PHYSICAL MODEL

In order to take the digital to the physical world, we opted to use resin 3D printing for the main structure. This technique is highly efficient in creating high fidelity 3D prints, which is exactly what was needed for our geometry. Although each of us had plenty experience in FDM printing, this was new to us and did require some iterations. Eventually requiring 3 tries to get the print to work for us. The settings that worked for us can be seen in Figure 28.

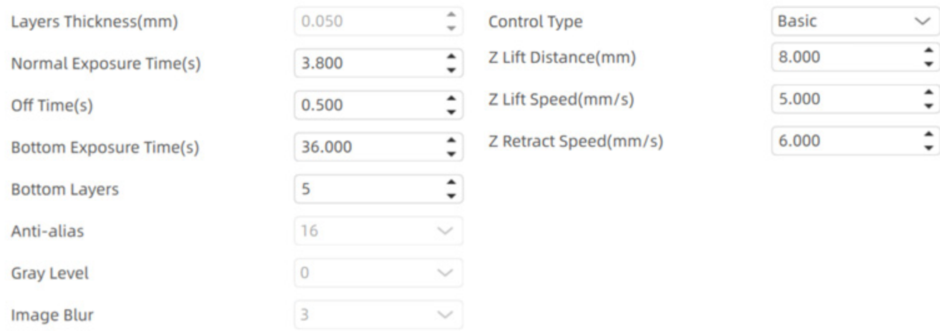


FIGURE 28 3D PRINTING SETTINGS

OWN WORK (2024)

What we discovered is that the bed needed to be levelled after each print, as the thousand(s) of cycles the printhead goes through does tend to move it a little. Furthermore the geometries need a larger baseplate than just the thin line of the projected bus stop. Additionally the supports the Anycubic slicer generates need some careful tweaking as they sometimes generate through the actual printable geometry (you can see the loss of two truss elements do to this failure in the actual model). Additionally we discovered that tall unsupported vertical elements can skew a little towards the top when printing, so either adding braced supports or dimensioning some 0,2-0,5mm tolerances between the structure and laser cut sheets would've prevented the need to manually sand the elements. Also, the washing and curing times of the print are highly dependable on the geometry. With normal prints usually washing and curing for 30 minutes each, our print was already a bit over-cured after 15 minutes. This made it slightly more brittle than we wanted too, however it did made removing the supports super easy. All in all we are quite pleased with the result.

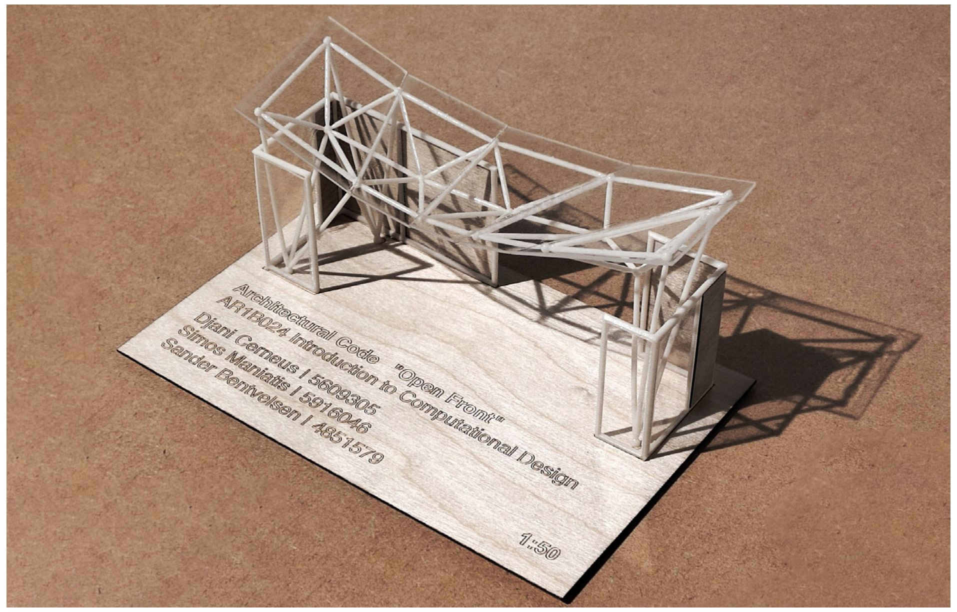


FIGURE 29 FINAL PHYSICAL MODEL

OWN WORK (2024)

REFLECTION

Djani Cerneus:

Exploring the realm of computation through the design of bus stops across Delft concisely encapsulates the essence of the course from my perspective. While I did not have a direct affinity for designing a bus stop structure, I derived immense satisfaction from experimenting with a diverse array of computational methods. In my view, the course exhibited a well-organised structure, and the weekly assignments on Miro were not only easy to follow but also served as an excellent ‘introduction to computational design’.

Collaborating with Simos Maniatis and Sander Bentvelsen was a delightful revelation, surpassing my expectations in teamwork. We seamlessly harmonised, avoiding conflicts and transforming the entire process into an enjoyable experience. Right from the start, our group established clear project objectives, attacking them with full force. In the project’s initial stage, we collectively devised a comprehensive workflow, providing a solid foundation for the remainder of the project.

Reflecting on the experience, I take pride in our team’s accomplishments. I am confident that we approached the project’s objectives properly by initiating the design process from scratch and by utilising computation not only for optimisation but also in a more abstract manner to understand the relationship between open and closed planes and their roles within space.

Simos Maniatis:

At its core, the script exemplifies the integration of data-driven design in architecture, in a post-parametric framework, since our goal from the beginning was not to design and then optimise but to generate and design through data. By using demographic data to determine the bounding size of the bus stop, the design process was grounded in the real needs and patterns of the users. This approach not only tailored the structure to its intended audience but also highlighted the potential of computational design in creating highly responsive and context-aware architecture.

However, the project was not without its challenges. Incorporating a multitude of design options for clients, while theoretically appealing, required a significant amount of precalculated simulations. This aspect of the project underscores the complexity inherent in computational design, where theoretical simplicity often meets practical complexity. The need for meticulous organisation to address bugs and manage the multitude of variables that needed to be addressed or accurately calculated.

Sander Bentvelsen:

During this course I learned more than I expected. Although I was already quite comfortable with the Grasshopper environment, it was refreshing to see the tutorials take a different approach to problems I would normally take, which made me reconsider and learn quite a few tricks. For me the tutorials themselves were structured brilliantly, sized exactly right to make it easy to navigate, sometimes skip over and sometimes rewatch whenever necessary.

What worked really well for us this project was the early approach to trying to integrating the entire design process. Using the flowchart methodology beforehand instead of after the design was made, we were able to lay out exactly where we wanted to go. Discussing what we wanted to implement and how we could do so. This resulted in the approach to defining positive and negative angles for the walls and roof. That allowed us to work on separate parts of the code that were later combined into one. In the end I’m incredibly happy with the group result and made some friends along the way. Each of us was enthusiastic in their own way and brought their own knowledge and stills to the table, which integrated pretty well.

What I perhaps would have liked to see in this course, is a little more freedom to the design assignments. I personally made separate files for the assignments and the actual group project, because they didn’t always overlap. Especially in the beginning of the course where we had to generate some random bus stop geometries. Perhaps it would’ve made the course more interesting if people that already know a bit of Grasshopper were incentivised to work ahead and integrate more varying plugins and use those as hand-ins.