Engineering Science & Technology
https://ojs.wiserpub.com/index.php/EST/

Research Article

Design of a Tiny House Generator with Location Parameterisation Function

Maximilian Frank Leon Schäfer, Stefan Schäfer, Nikola Bisevac*

Institute of Constructive Design and Building Construction, Technical University of Darmstadt, Darmstadt, Hessen, Germany
E-mail: bisevac@kgbauko.tu-darmstadt.de

Received: 3 July 2025; **Revised:** 5 September 2025; **Accepted:** 23 September 2025

Abstract: In light of growing housing shortages and rising rental prices, alternative housing forms such as Tiny Houses are becoming increasingly popular. This housing form is characterised by compact floor plans, with sizes typically under 45 m^2 (480 sqft) per person. As the trend originated in the USA, much of the literature and design proposals refer to the prevalent climate conditions found there. This research aims to bridge this gap by developing a script that generates a proposal for a Tiny House for any given location, using construction strategies adapted to the local climate. First, an analysis was conducted to determine which building components of a Tiny House are particularly susceptible to climatic influences and how specific weather conditions affect these components. Based on four case studies and relevant literature, parametric construction principles were developed. These principles were incorporated into a script that used weather and climate data to generate a 3D model of the Tiny House. The script was implemented within the Grasshopper environment of the 3D modelling software Rhino 3D. To provide a user-friendly interface, it was integrated into a web application. This allows users to select locations and various input parameters, to visualize the generated model, as well as to access detailed information about the construction decisions and how they are influenced by the local climate. To exemplify the output generated by the tool, three models for different locations were selected and slightly modified to show how these buildings might be built and look in reality. The thesis was successful in developing a fully parametric building generator, which can further be expanded to include features such as complete indoor climate simulations. The script and implementation are fully documented. However, given the general complexity of architecture and construction, the question arises as to whether a future approach based on artificial intelligence might be more effective than the algorithmic approach taken here.

***Keywords*:** parameterisation, tiny house, micro house, Rhino, Grasshopper, climate-based design, parametric design, automated design

1. Introduction to tiny houses

In recent years, urban centres around the world have been confronted with a growing housing shortage and steadily rising rents. At the same time, climate change is increasing the urgency of sustainable housing concepts. In this context, alternative forms of housing such as tiny houses are becoming increasingly important. These small, efficiently designed living spaces not only offer an answer to the lack of space and high housing costs but also provide an opportunity to reduce the ecological footprint.

DOI: https://doi.org/10.37256/est.7120267689

A clear definition of tiny houses does not yet exist, but the term generally refers to small residential structures. In order to find a definition for this work, legal texts are consulted. For example, the German “Hesse Housing Supervision Act” stipulates a minimum square metre requirement for rental flats of 9 m^2 per person [1]. Common definitions of tiny houses in the literature state around 40 m^2 (400 sqft). According to these definitions, a tiny house must be fully equipped with a kitchen, bathroom, dining area, living area, sleeping area, and storage space [2]. In [3], Mitchell adds the number of people living in the household to the definition. According to this source, a house for a family of five with 90 square metres can also be described as a tiny house. A house between 9 and 40 m^2 per person will therefore be defined as a tiny house.

The tiny house movement originated in the United States of America, where a steady increase in living space per person has been observed since the 1970s. Sizes have doubled from an average of 50 m^2/pp to 100 m^2/pp [4], [5]. This development is in direct contrast to the idea of tiny houses, which emerged as a counter-movement to the trend of ever larger houses. Given the small size available, this form of living is characterised by efficiency and a minimalist lifestyle that aims to limit living space to the essentials and make optimal use of it [2], [6]. The idea of living in small spaces and minimalism is not a new concept. Even in ancient times, the philosopher Diogenes of Sinope is said to have lived in a barrel and only owned the bare essentials [7]. Examples such as early American settler houses (around 15 m^2 [8]) or current Japanese living conditions (∅ 23.3 m^2/pp [9]) also show that small living spaces were and still are common in various cultures and eras [6].

Modern Tiny Houses have various applications, from hotel resorts and holiday accommodation to private retreats or even main residences. It is widespread in the tiny house community to build your own tiny house as a Do It Yourself (DIY) project. Economic factors and independence are often the decisive motivations for such projects. As Tiny Houses are cheaper than traditional residential buildings and use fewer materials and are less complex due to their efficient design, they can also be realistically realised by non-professionals [2], [3]. Tiny houses are not only built and used by young people. One example of other types of use is the Chandler Boulevard Tiny House Village in Los Angeles, a project that uses tiny houses as a solution to homelessness. By creating small, affordable living spaces, people without a permanent home are given the opportunity to provide safe and dignified accommodation. Furthermore, Tiny Houses can be a good form of housing for “Empty Nesters”, i.e., people whose children have left home, but also for students or older people who prefer an independent lifestyle. They are also suitable as home offices or guest houses [10].

Despite the growing interest and diverse applications of Tiny Houses, there are only a few works that deal with construction principles for regions outside the USA. Since most of the literature focuses on the climatic and structural conditions of the US, there is a gap in information on other geographic and climatic conditions. This gap makes it necessary to develop new approaches that identify specific requirements for different environmental conditions. This thesis investigates whether parametric design approaches can be used to demonstrate the design principles of Tiny Houses in different environmental conditions. The focus is on the development of a script that generates designs based on specific environmental parameters such as climate and sun exposure. Firstly, the basic characteristics of Tiny Houses are presented, followed by an examination of current trends. Methods generating tiny houses using parametric methods are then discussed. The insights gained could be relevant for those responsible for building and those interested in Tiny Houses, as they can serve as a guideline for making informed decisions and finding sustainable solutions for the construction of Tiny Houses.

1.1 *Central components of a tiny house*

In order to develop a design for tiny houses that can be adapted to environmental parameters, it is first necessary to identify the basic components that make up a tiny house. These core design features then form the basis for the development of adaptation strategies. According to [11], the components of a building can generally be divided into separating and load-bearing components. These include floors, roofs, columns, external walls, internal walls, doors, windows, balconies, and stairs. Whilst some Tiny Houses have a gallery or a sleeping loft, most designs are single-storey only. Therefore, balconies and stairs are not relevant for tiny houses while the following components are particularly important.

1.1.1 *Supporting structure*

The supporting structure of a building is the fundamental component of a construction (Figure 1). Load-bearing components are independent of the subordinate components, so the load-bearing structure can exist without the other components, but not vice versa. In detail, the supporting structure comprises the main structural elements such as columns, beams, and load-bearing walls. These elements are arranged that they take up the loads from the entire building, including their own, and transfer them to the foundation [11]. In addition, the supporting structure plays a central role in buildings as a thermal mass. By storing and releasing thermal energy (or cold), it can influence the indoor climate [12]. The higher the thermal mass, the greater the energy storage capacity. Different load-bearing systems have different thermal masses due to the materials they are made of and their specific thermal conductivity and heat capacity.

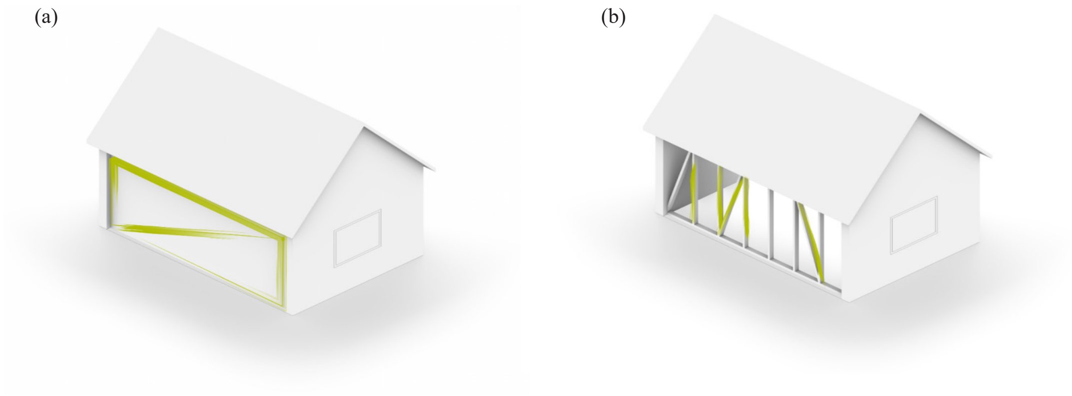

Figure 1. Supporting structures illustrated—monolithic (a); lightweight construction (b)

The choice of supporting system is therefore influenced by the climatic conditions, among other things. Correct application of the principles of thermal mass is particularly important for tiny houses, where thermal mass is limited [13]-[16].

1.1.2 *Facade*

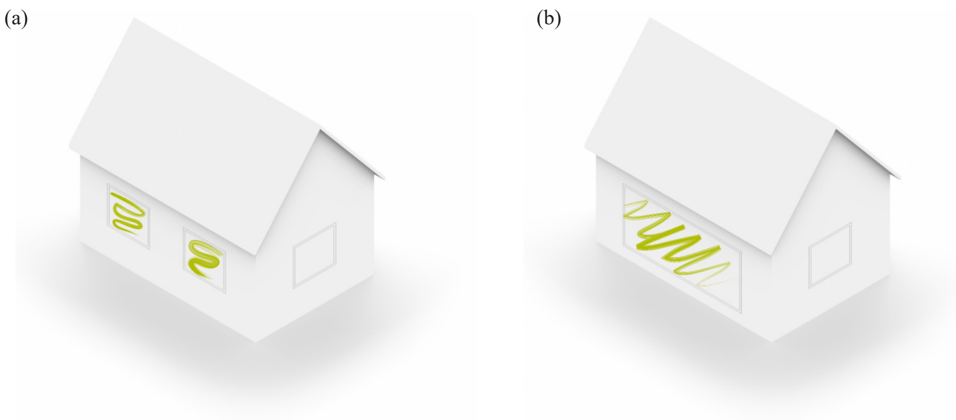

Figure 2. Window to Wall Ratio (WWR) illustrated—south façade with 0.2 WWR (a); 0.4 WWR (b)

The façade of a building not only performs the essential task of weather and temperature protection, but also significantly defines the external appearance of the building. A façade system can be part of a load-bearing component or can also be merely a building envelope (Figure 2). Façades also contain windows, which are responsible for the transfer of sunlight and ventilation. The number, size and position of the windows influence the amount of daylight entering the building and therefore the energy required for artificial lighting. An optimal window arrangement can reduce the need for artificial lighting and at the same time maximise or minimise heat gains from solar radiation, depending on the climatic conditions and the orientation of the building. As the façade together with the number, size, and position of windows (WWR) significantly influences indoor comfort and energy efficiency of the building, parametric adaptability also makes sense here [11], [17].

1.1.3 *Solar*

The use of solar panels is widespread in the Tiny House community. Many residents value environmental friendliness and independence. Solar systems also enable so-called "off-grid living", where a house is not connected to the public power grid. In conjunction with battery storage systems, the electricity demand can be covered throughout the year. Due to the small living space and the resulting reduced energy requirements, tiny houses are particularly suitable for operation with Photovoltaic (PV) systems. The parameterisation of solar systems offers an important optimisation option. By simulating the tilt and orientation of the solar modules (Figure 3), various configurations can be tested in order to maximise the energy yield while taking site conditions into account [2], [18].

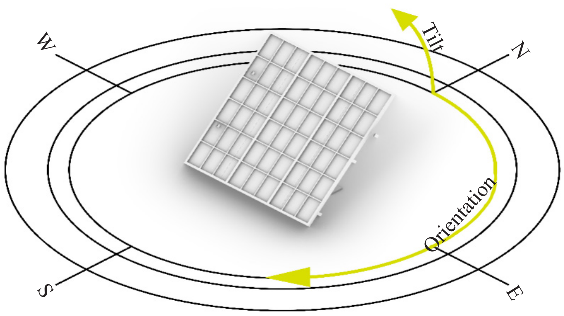

Figure 3. Tilt and orientation of solar modules illustrated

1.1.4 *Roof*

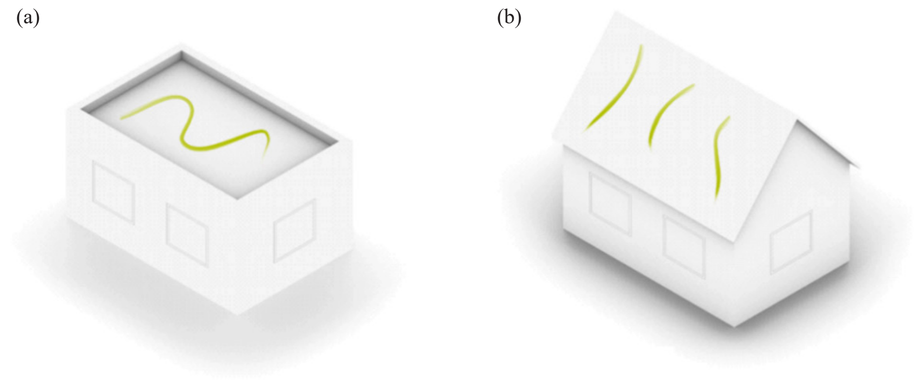

Figure 4. Differentiation of roof types illustrated—flat roof (a); pitched roof (b)

In addition to the exterior walls, the roof of a building is another separating component that fulfills several central functions (Figure 4). Like the façade, it provides protection from the weather, drains rainwater, and, as a separating component, must also protect against the effects of temperature. Roofs can be categorised into two main types: flat roofs and pitched roofs. The choice of roof type not only depends on aesthetic considerations but also on external factors such as wind strength, snow load, and the amount of precipitation that occurs at the building's location [11], [15]. Flat roofs are roofs with a low pitch of less than 10 degrees. A flat roof allows the roof to be used as an additional surface, for example, as a roof terrace, a garden, or for the simple installation of solar panels. In general, the drainage of precipitation is a major weakness of flat roofs. Flat roofs should also be avoided in regions with high winds, as the suction effect on flat roofs is strong and there is a risk of the roof lifting off. Pitched roofs offer a better solution for regions with high rainfall and snow loads. The angle of inclination enables effective drainage of rainwater and prevents the accumulation of snow. Another advantage of pitched roofs is the possibility of integrating roof overhangs and canopies, which provide additional protection from the sun and rain [19], [20].

1.2 *Types of foundations*

There are three different types of Tiny Houses, which are mainly differentiated by their foundation types: Tiny Houses on Wheels (THOW), Tiny House on Foundation (THOF), and Tiny Houses on Skids (THOS) [2] (Figure 5). Finally, in some cases, houseboats are also categorised as tiny houses [6].

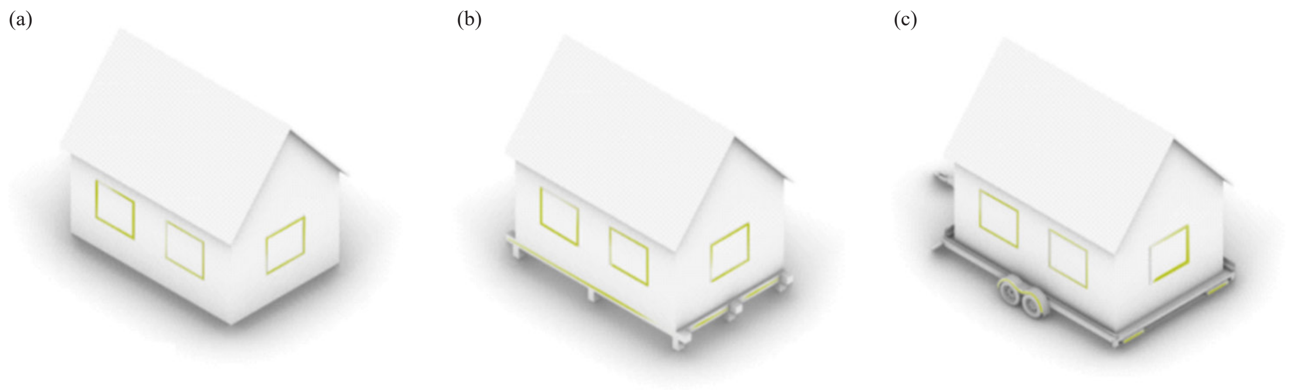

Figure 5. Foundation types illustrated—THOF (a), THOS (b), THOW (c)

2. Influence of climatic conditions on design

Every optimised construction is the product of the combined consideration of all its boundary conditions [21]. For example, a house that is optimised for a location in Sweden would be constructed differently from a house that is to be built in Spain. It is to be expected that in the former, a higher value must be placed on protection against the cold in winter, while in the latter, heat protection in summer is more important. The two locations therefore have at least different thermal boundary conditions. It is therefore necessary to determine the different boundary conditions that influence the construction of a building. According to Lippsmeier in [15], the following location-based boundary conditions can be distinguished:

- Temperature;
- Precipitation;
- Air humidity;
- Solar radiation;
- Wind.

A key feature of the Tiny House Generator is its ability to adapt the design to the local environmental conditions to propose an optimised design. Once the components that are to be parametrically adapted have been identified, it is necessary to identify the characteristics of the environment, which vary according to location, in order to develop appropriate design strategies. This would require cataloguing all possible locations for the placement of the Tiny House.

Such a complex, global categorisation of different environmental conditions is usually carried out using climate maps or databases linked to Geographical Information Systems (GIS) [22].

2.1 *Temperature and precipitation*

The Köppen-Geiger climate classification is a widely used method for classifying climate zones that was first developed by Wladimir Köppen in 1900 and later further developed by Rudolf Geiger in 1961. This classification is based on the average temperatures over the annual cycle and on precipitation. A distinction is made between five main climate zones [23], [24]:

- Tropical climate (A);
- Dry climate (B);
- Warm temperate climate (C);
- Snow climate (D);
- Polar climate (E).

The Köppen-Geiger climate classification is based on defined threshold values for temperature and precipitation, which are shown in detail in Table 1. The classification is letter-based and organised in a tree structure. Figure 6 shows a map with the Köppen-Geiger climate classification. The colours of the climate zones were taken from [25].

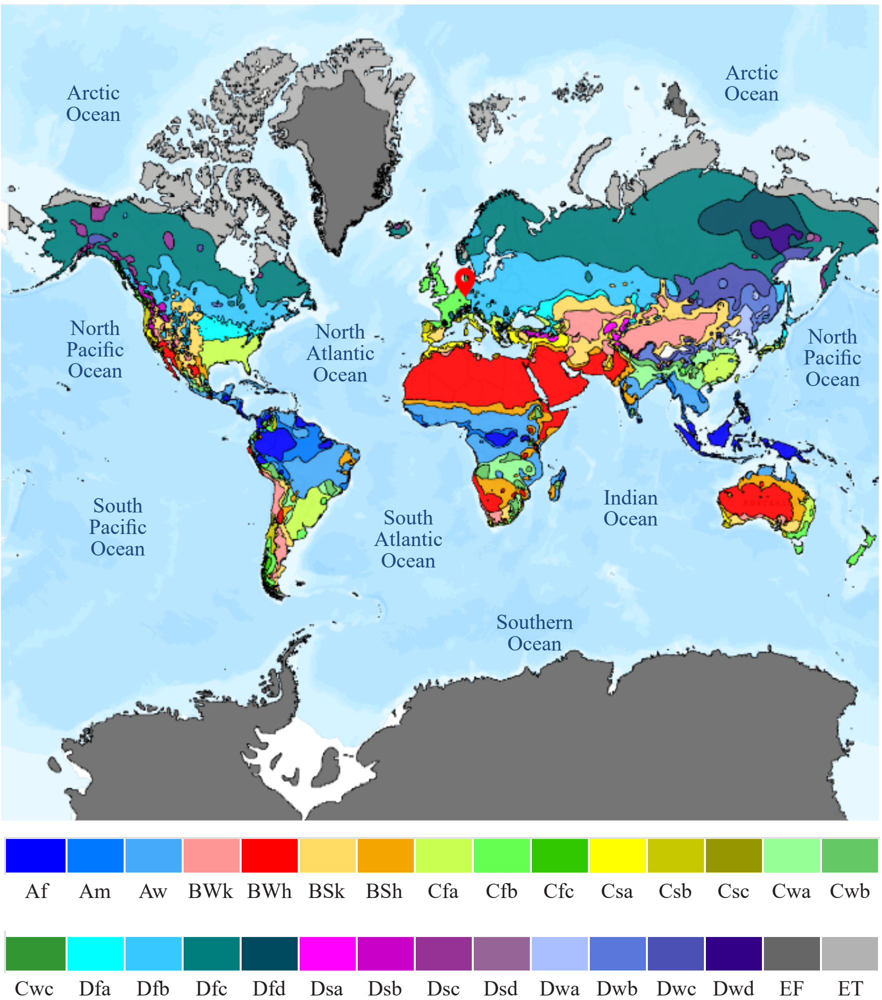

Figure 6. Köppen-Geiger climate map with legend

Table 1. Köppen-Geiger classification criteria [22], [23], [25]

Classification			Description	Threshold value	
Primary	Secondar	Tertiary		Temperature	Precipitation
A			Equatorial climate	$T_{min} \geq +18$ °C	
	f		Rainforest, completely humid		$P_{min} \geq 60$ mm
	m		Monsoon		$P_{ann} \geq (100\text{ mm} - P_{min})/25$
	w		Savannah with dry summers		$P_{ann} < (100\text{ mm} - P_{min})/25$
B			Dry climate zones		$P_{ann} < 10P_{th}$
	S		Steppe climate		$P_{ann} > 5P_{th}$
	W		Desert climate		$P_{ann} \leq 5P_{th}$
		h	Hot steppe/desert	$T_{ann} \geq +18$ °C	
		k	Cold steppe/desert	$T_{ann} < +18$ °C	
C			Warm temperate climate	-3 °C $< T_{min} < +18$ °C	
	s		Warm temperate climate with dry summers		$P_{smin} < P_{wmin}$ & $P_{wmax} > 3P_{smin}$ & $P_{smin} < 40$ mm
	w		Warm temperate climate with dry winters		$P_{wmin} < P_{smin}$ & $P_{smax} > 10P_{wmin}$
	f		Warm temperate climate, completely humid	Neither Cs nor Cw	
		a	Hot summer	$T_{max} \geq +22$ °C	
		b	Warm summer	Neither a and at least $4T_{mon} \geq 10$ °C	
		c	Cool summer and cold winter	Neither b and $T_{min} > -38$ °C	
D			Snow climate	$T_{min} \leq -3$ °C	
	s		Snow climate with dry summers		$P_{smin} < P_{wmin}$ & $P_{wmax} > 3P_{smin}$ & $P_{smin} < 40$ mm
	w		Snow climate with dry winters		$P_{wmin} < P_{smin}$ & $P_{wmax} > 10P_{wmin}$
	f		Snow climate, completely humid	Neither Ds nor Dw	
		a	Hot summer	$T_{max} \geq +22$ °C	
		b	Warm summer	Neither a and at least $4T_{mon} \geq 10$ °C	
		c	Cool summer and cold winter	Neither b and $T_{min} > -38$ °C	
		d	Very cold winter	Like c but $T_{min} \leq -38$ °C	
E			Polar climate	$T_{max} < +10$ °C	
	T		Tundra	0 °C $\leq T_{max} < +10$ °C	
	F		Frost climate	$T_{max} < 0$ °C	

Based on the Köppen-Geiger climate map, the city of Darmstadt (red pin), for example, could be assigned to climate zone Cfb. This zone is characterised by a warm temperate climate with year-round humidity and warm summers. The average temperatures in winter are between -3 °C and 18 °C, with the temperature rarely falling below -3 °C even in the coldest months. In the summer months, the temperature often reaches over 10 °C, with average temperatures above this threshold in at least four months of the year. Rainfall is evenly distributed throughout the year, with no distinct dry or rainy seasons.

2.2 *Influence of solar radiation*

The sun plays a decisive role in the well-being of people in buildings. It has long been recognised in architecture that people are positively influenced by light and warmth. These basic psychological needs are best fulfilled by natural light, and this is difficult to imitate artificially. Only a room with natural lighting through windows offers a direct connection to nature and the environment. In addition, sunlight contributes to energy savings in terms of lighting, heating, and solar energy [26], [27]. As mentioned in the Chapter 1.1, the properties of the sun have a considerable influence on the physical and architectural design of buildings. The direction and intensity of sunlight have an impact on the dimensioning of shading devices and windows as well as the orientation of buildings, the positioning and yield of solar collectors and the thermal load caused by solar radiation [28]. Buildings in colder climate zones and at higher latitudes are often designed to maximise the use of solar radiation, while buildings in hot regions are often designed to avoid direct sunlight and the associated heating of rooms [26]. Planning a building requires an understanding of the sun's path, which varies according to latitude and season [29] (Figure 7).

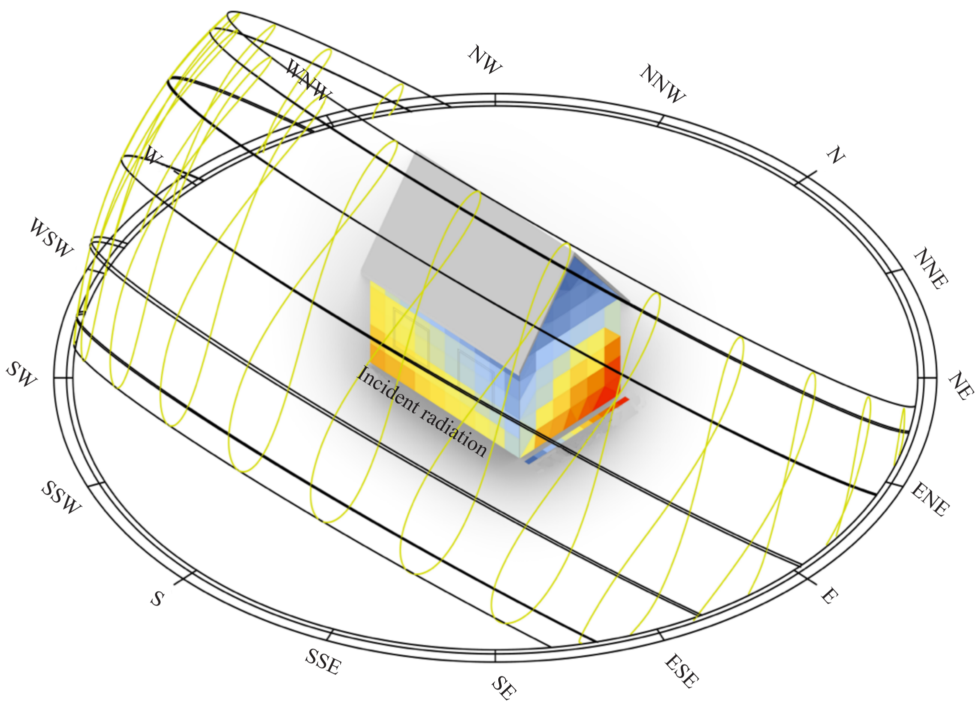

Figure 7. Solar simulation illustrated on a façade

3. Parameterisation of the components

When the design process of a building interacts closely with the environmental conditions in which it is erected, this is referred to as climate-based design [30]. However, according to the Köppen-Geiger classification alone, 30 different climate zones can be identified worldwide. Theoretically, it would be possible to develop a structurally optimised design for each of these climate zones. However, if the position of the sun is also analysed or if a case-by-case calculation of the façade is considered, this results in an almost endless number of possible combinations. On the other hand, a universal solution that is equally suitable for all climate zones is interesting given the current state of technology, but probably neither economical nor efficient. This diversity makes it impossible to model and analyse all variants manually. This is where a parametric approach can prove useful. However, the aforementioned climate classifications can still be helpful when selecting materials and construction methods [31].

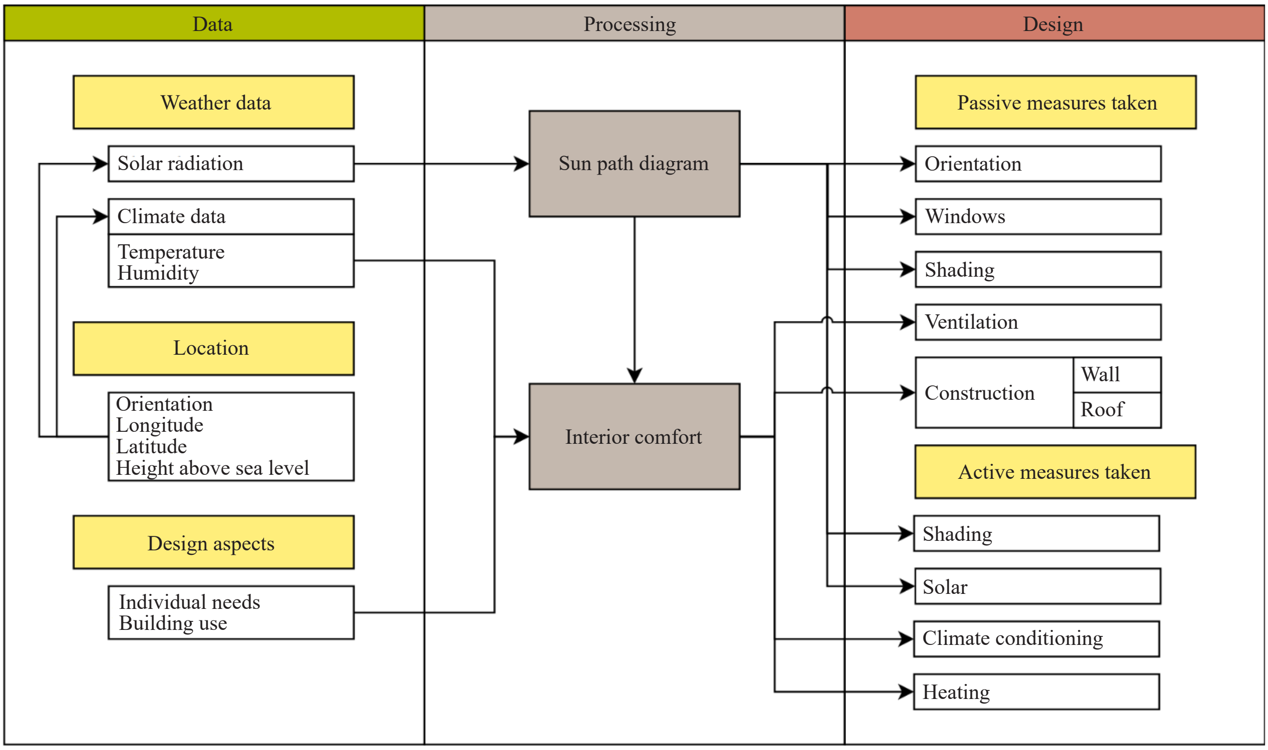

Figure 8. Climate-based design based on the Input-Processing-Output (IPO) principle according to [15]

Figure 8 illustrates the basic process of climate-related design, presented according to the IPO principle [32], [33], a common form of representation for data processing in software engineering [15], [30]. The process begins with the collection of meteorological and climatic data, such as information on solar radiation, temperature and humidity, which is usually linked to location data. This also includes information on geographical features and topographical characteristics, which are incorporated into the design. In addition, the individual requirements of the user are usually taken into account during the planning process. The collected data is then analysed and interpreted. For example, a sun position analysis based on meteorological data and location data can provide information about the sun's path throughout the year. The planner considers how indoor comfort is affected in terms of temperature, ventilation and lighting conditions, such as the orientation of windows to maximise daylight or minimise direct sunlight in summer. The third and final step is the development and adaptation of specific building measures based on the analysis of the data collected. Architectural measures are divided into passive and active categories. Passive measures include design or construction elements that function without the use of technical systems and do not require additional energy sources. Examples include ventilation through windows or the placement of overhangs for shading. As they are more energy efficient over the entire life cycle of a building, passive measures should generally be favoured [34]. Active measures, on the other hand, rely on technical systems, such as mechanical shading or heating and cooling systems, which require additional energy sources.

The procedure shown in the first diagram encounters challenges in practical implementation in the context of a parametric algorithm. A conventional algorithm works sequentially and can only make decisions in one direction. However, a problem arises with mutually dependent aspects. For example, if the shading is adjusted to reduce solar heat gains, the gains through the windows changes, which in turn makes it necessary to adjust the window size. These interactions make it necessary to run through the process several times. This also raises the problem of prioritisation: which measures should be taken first? For example, should the shading be optimised first or should the window size be adjusted first in order to minimise solar gain? For such problems, a genetic algorithm is often used, which is able to optimise complex interactions between several parameters towards a desired value [35]. A variety of possible solutions are tested iteratively in order to find an increasingly better optimised combination, which then attempts, for example, to maximise the window area while minimising the solar heat gain. One disadvantage of genetic algorithms, however, is that they are time-consuming and computationally expensive. As many different possible solutions are checked and evaluated in each iteration step, the process takes considerably longer than with conventional algorithms. The calculation effort can be extremely high, especially for complex tasks, such as a building, which contains many parameters and interactions. Some adjustments must therefore be made for implementation as an algorithmic linear programme. These adjustments are shown in Figure 9.

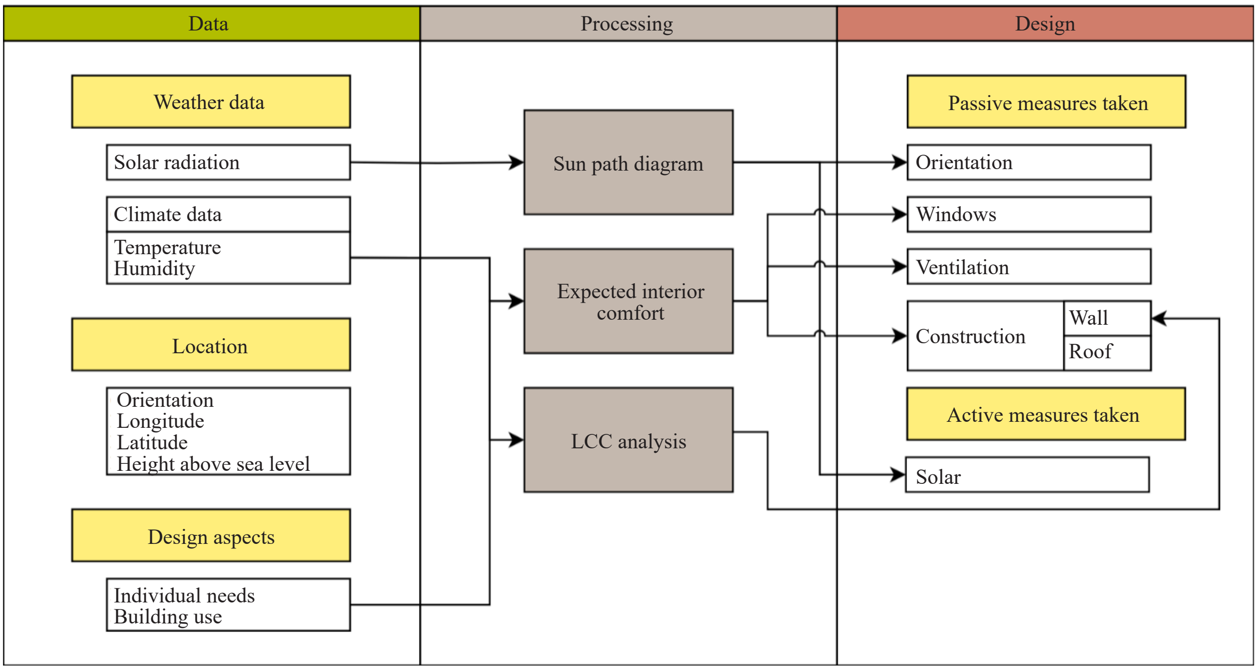

Figure 9. Climate-based design adapted for linear algorithm according to the IPO principle

Initially, individual user requirements cannot be set directly, as the programme is designed to always develop an optimised solution for the specific climate zone. General design principles common to the zone are therefore used. Another obstacle is the detailed simulation of indoor comfort. Due to the complexity and the considerable computational effort involved, it is not possible to carry out a comprehensive comfort simulation in this programme. Instead, expected, standardised values are used. A more detailed analysis can be carried out for the façade by using a Life Cycle Cost Analysis method (LCCA method, see Chapter 3.2.1) [36], [37]) to determine the optimum insulation thickness in relation to the costs. The final design also largely dispenses with active measures. On the other hand, the simulation of solar energy generation and the orientation of the PV remain feasible.

3.1 *Analysis of case studies*

To capture the state of the art in the construction and use of tiny houses, four exemplary projects from different climate zones were selected and analysed on the basis of the components in focus. The selected projects are Casa Sustentável in Brazil, Casa Caldera in the USA, ÁPH80 in Spain, and the Tochka na Karte project in Russia (Table 2).

Table 2. Selection of case studies

Project		Climate zone
Casa sustentável	Aw	Savannah with dry summers
Casa caldera	BSh	Hot steppe
ÁPH80	Cfb	Warm temperate climate, completely humid, warm summer
Tochka na karte	Dfc	Snowy climate, completely humid, cool summer and cold winter

3.1.1 *Casa sustentável*

The Casa Sustentável project, a 45 m^2 tiny house, was built by Gustavo Penna Arquiteto & Associados in Minas Gerais, Brazil, using by-products from mining. It is located in climate zone Aw (savanna with dry summers) (Figure 10). The walls are made of solid masonry and have no additional insulation, as the thermal mass of the structure itself regulates the interior temperature. Cooling is provided by the thermal mass, while heating is supported by solar energy. A particularly interesting feature is the shading provided by the canopy. This was dimensioned so that the high summer sun does not reach the interior of the room. However, the lower winter sun shines through the shading and is used for solar heat gain [38].

Figure 10. Impressions of the casa sustentável case study
Source: https://www.gustavopenna.com.br/casasustentavel

3.1.2 *Casa caldera*

Casa Caldera, designed by DUST, is located in Arizona, USA, and has a floor area of 49 m^2 (Figure 11). This part of Arizona is in climate zone BSh (hot steppe). The load-bearing structure is made of solid concrete (lavacrete). Here, too, insulation is not used, as the solid construction provides sufficient thermal inertia to compensate for extreme temperature fluctuations. A large central corridor can be opened for night ventilation. Heating is provided by a wood-burning stove when needed. Solar energy is used as the main energy source. The windows are kept remarkably small, presumably to minimize solar radiation and thus keep the interior cool [38].

Figure 11. Impressions of the casa caldera case study
Source: Quelle: https://dustarchitects.com/endeavors/casa-caldera/

3.1.3 *ÁPH80*

The ÁPH80 project in Huesca, Spain, was designed by Ábaton Arquitectura and, at 27 m^2, is the smallest of the projects considered (Figure 12). Huesca, near the Pyrenees, is located in climate zone Cfb (warm temperate climate, completely humid, warm summer), which is characterized by mild temperatures. This tiny house uses a mixed construction method consisting of a wooden frame, wood-concrete panels, and a ventilated insulation system. Compared to the other projects, the house is transportable (THOS) and must be connected to an external energy source on site. Large windows, which can be easily shaded, allow maximum use of natural light and contribute to passive solar heat gain [7].

Figure 12. Impressions of the ÁPH80 case study
Source: https://abaton.es/en/projects/portable-home-aph80/

3.1.4 *Tochka na karte*

The Tochka na Karte tiny house project, designed by RHIZOME, is located in Priozersk, Russia, and has approximately 37 m^2 per residential unit (Figure 13). It is a group of individual hotel apartments on Lake Ladoga. This area is located in climate zone Dfc (snow climate, completely humid, cool summers and cold winters). The supporting structure is made of wood, a material with low thermal mass. These materials are attractive for cold climate zones because such buildings can be heated quickly and more energy-efficiently [13]. A thick, rear-ventilated insulation system provides protection from extreme climatic conditions. Heating is provided by an unspecified external energy source. The particularly large windows on the south side are striking, allowing plenty of daylight into the rooms [39].

Figure 13. Impressions of the tochka na karte case study
Source: https://www.archdaily.com/891641/tochka-na-karte-hotel-rhizome

3.1.5 *Comparison*

The four projects show different approaches to adapting to their respective climate zones (Table 3). In tropical and hot climate zones (Casa Sustentável and Casa Caldera), the focus is on thermal mass and passive cooling to ensure a pleasant indoor climate. In temperate climate zones (ÁPH80), greater emphasis is placed on insulation systems and generous ventilation options. In subarctic climates (Tochka na Karte), the focus is increasingly shifting towards insulation, sunlight and “warm” materials.

Table 3. Comparison of the case studies [7], [38], [39]

Project	Casa sustentável	Casa caldera	ÁPH80	Tochka na karte
Architect	Gustavo Penna Arquiteto & Associados	DUST	Ábaton Arquitectura	RHIZOME
Size	45 m^2	49 m^2	27 m^2	37 m^2
Position	Minas Gerais, Brazil	Arizona, USA	Huesca, Spain	Priozersk, Russia
Climate zone	Aw	BSh	Cfb	Dfc
Supporting structure	Solid, masonry	Solid, concrete	Mixed construction	Wood
Insulation	None	None	Ventilated insulation system	Ventilated insulation system
Cooling	Thermal mass	Cross ventilation	Ventilation	Ventilation
Heating	Solar heating	Wood stove	n/a	Connection from outside
Energy concept	Renewable energies	Solar energy	Connection from outside	Connection from outside
Sunlight	Shading optimised according to the season Position of the sun	Small windows	Large windows	Large windows

3.2 *Development of design principles*

The case studies in Chapter 3.1 confirm the assumption that the basic design principles of tiny houses are comparable to those of larger buildings. When considering indoor comfort, familiar concepts such as thermal insulation, thermal mass, ventilation and sunlight play a central role in both cases. Nevertheless, tiny houses place special demands on the building envelope, as the overlapping functions of the rooms require better control of the indoor climate. For example, there are different requirements for sleeping and living areas, but these can be spatially overlapping in a tiny house. In particular, ventilation, shading, and technical building equipment must be sufficiently dimensioned to allow for different climatic conditions within the same room [2], [13], [14], [16].

It is therefore particularly important to apply the correct construction principles to Tiny Houses. The key principles for wall construction, PV, roof design, and shading are discussed in detail below.

3.2.1 *Wall construction and insulation*

In addition to the load-bearing capacity, temperature and humidity are the main factors influencing the choice of wall construction. These two parameters are also the basic characteristics of the Köppen-Geiger climate classification from Chapter 2.1. This classification categorises climate zones based on temperature and precipitation patterns and thus provides a valuable basis for adapting structural decisions to the respective climatic conditions. The following section explains how the wall construction should be varied depending on the climatic requirements.

3.2.1.1 *Influence of temperature on the wall construction*

Depending on the climatic conditions on site, wall construction must provide either heat and/or cold protection. The choice of a suitable construction method depends on the expected temperatures, which are categorised into three main climate zones in the Köppen-Geiger classification:

• Equatorial climate zone (more than 18 °C average temperature): In tropical climate zones, the focus is on thermal insulation, as protection against the cold is not required. The wall construction must primarily protect the interior from excessive heat. There are differing opinions as to whether insulation is necessary in these climate zones. Some sources recommend solid, monolithic constructions without insulation, while others rely on an insulating layer to reduce heat input [40]. In this work, a calculation for the most economical insulation thickness is applied.

• Temperate climate zone (-3 °C to 18 °C average temperature): In temperate climate zones, both cold and heat insulation must be taken into account. A monolithic construction method with an additional insulation system is used here. The chosen construction is a solid interior wall with an External Thermal Insulation Composite System (ETICS), inspired by ÁPH 80.

• Cold climate zone (below -3 °C average temperature): In cold climate zones, ranging from snowy climates to polar climates, protection against the cold takes centre stage. Here, a skeleton construction method, such as timber frame or steel construction, is often combined with a highly effective insulation system. This construction method provides sufficient thermal insulation to protect the interior from extreme cold, while the low thermal mass of the lightweight construction allows it to be heated easily, similar to the project Tochka na Karte.

• Special requirements for THOW: If the Tiny House is to be mobile, the wall construction must be of skeleton construction for weight reasons. The maximum weight of the building is severely limited by the maximum trailer load.

3.2.1.2 *Influence of moisture on wall construction*

In addition to temperature, humidity is also a major challenge for the wall construction. Depending on the climatic humidity, different façade systems are used [41], [42]:

• Humid climates: In regions with high humidity, a rear-ventilated façade makes sense. This construction method prevents moisture from accumulating in the walls through constant air circulation behind the façade.

• Medium and dry climates: In dry or moderately humid climates, a monolithic façade is often used. This construction method is cheaper in terms of initial investment and takes up less space than a ventilated façade. As the moisture load is lower, the construction requires less elaborate protective measures against moisture.

3.2.1.3 *Calculation of the optimum insulation thickness*

Calculating the optimum insulation thickness for a building is a complex task that goes far beyond the simple "more insulation is better" approach. Although greater insulation reduces heat loss and the need for heating in winter and cooling in summer, construction costs also rise with increasing insulation thickness. This leads to a point where the additional savings in energy costs no longer justify the higher investment costs. One approach to this problem is a Life Cycle Cost Analysis (LCCA). This method is based on a cost-benefit analysis that considers the life cycle costs of a building [36], [37], [43]. In particular, the LCCA offers sufficient computational simplicity for use in an algorithm such as that of the Tiny House Generator, compared to more complex energy-related simulations [37], [44]. Heating Degree Days (HDD) and Cooling Degree Days (CDD) are used to assess the thermal effects of the climate on the interior and to calculate the heating and cooling requirements. These metrics provide information on how many degrees per day the outside temperature deviates from a defined reference temperature, which can be used to estimate the annual heating and cooling requirements of a building [37], [45].

• Heating Degree Days (HDD) are calculated by summing the difference between the average outdoor temperature and a defined base temperature over a certain period of time. This metric is used to determine the annual heating requirement in a specific climate. The higher the HDD values, the more energy is required for heating [37].

• Cooling Degree Days (CDD) are the sum of the daily average temperatures that are above a defined base temperature. This metric is used to determine the annual cooling requirement. Higher CDD values indicate a greater need for cooling [37].

The HDD and CDD can be derived from weather data such as EnergyPlus Weather (EPW) files. These contain hourly values for temperatures at a location (see Chapter 4.3). In this work, the following standard values are selected for the base temperatures: 18 °C for the heating requirement and 23 °C for the cooling requirement. These values correspond to common assumptions in building planning and also standard values in many simulation tools. Other standard values and the following formula are taken from [37].

$$x^{\text{opt}}_{\text{heat-cool, wall}} = \left(0.024 \frac{HDD \cdot C_{\text{elec}} \cdot PWF \cdot \lambda}{C_{\text{insu}} \cdot \eta} + 0.024 \frac{CDD \cdot C_{\text{elec}} \cdot PWF \cdot \lambda}{C_{\text{insu}} \cdot \varepsilon} \right)^{1/2} - R^{\text{actual}}_{\text{wall}} \cdot \lambda$$

Table 4 lists the variables required for calculating the insulation thickness based on the Life Cycle Cost Analysis (LCCA) method.

Table 4. Variables in the calculation of the insulation thickness according to the LCCA method [36], [37]

Variable	Description	Value
$x^{\text{opt}}_{\text{heat-cool, wall}}$	The optimum wall insulation thickness based on the HDD and CDD in metres	
$R^{\text{actual}}_{\text{wall}}$	Total thermal resistance of the wall structure without insulation	
PWF	Present value factor (adjusts future energy costs to their present value)	21.1
η	Efficiency (Seasonal Coefficient of Performance (SCOP)) of the electric heat pump	2.50
ε	Energy efficiency ratio (Seasonal Energy Efficiency Ratio (SEER)) of the electric heat pump	3.00
C_{elec}	Electricity price	0.2236 €/kWh
C_{insu}	EPS insulation costs	263.78 €/m^3
λ	Thermal conductivity of EPS insulation	0.034 W/mK

3.2.1.4 *Windows*

The thermal performance (U-Value) of the windows is not adjusted parametrically; instead, it is selected analogous to that of the façade. The parameters are primarily concerned with the orientation, number and size of the windows, in particular the "Window-to-Wall Ratio" (WWR). The WWR describes the ratio of the window area to the total façade area and significantly influences the solar heat input, but also the ventilation of the building.

Table 5. WWR by climate zone and direction

Klimazone	WWR north	WWR east	WWR south	WWR west
Af	0.4	0.4	0.25	0.4
Am	0.4	0.4	0.25	0.4
Aw	0.4	0.35	0.25	0.35
BSh	0.4	0.35	0.25	0.35
BSk	0.4	0.4	0.56	0.4
BWh	0.4	0.35	0.25	0.35
BWk	0.4	0.4	0.56	0.4
Cfa	0.37	0.33	0.27	0.33
Cfb	0.43	0.39	0.4	0.41
Cfc	0.4	0.4	0.56	0.4
Csa	0.37	0.33	0.27	0.33
Csb	0.43	0.39	0.4	0.41
Csc	0.4	0.4	0.56	0.4
Cwa	0.37	0.33	0.27	0.33
Cwb	0.37	0.33	0.27	0.33
Cwc	0.4	0.4	0.56	0.4
Dfa	0.43	0.39	0.4	0.43
Dfb	0.45	0.4	0.56	0.45
Dfc	0.45	0.5	0.7	0.5
Dfd	0.45	0.5	0.7	0.5
Das	0.4	0.4	0.56	0.4
Dsb	0.4	0.4	0.56	0.4
Dsc	0.45	0.5	0.7	0.5
Dsd	0.45	0.5	0.7	0.5
Dwa	0.4	0.4	0.56	0.4
Dwb	0.4	0.4	0.56	0.4
Dwc	0.45	0.5	0.7	0.5
Dwd	0.45	0.5	0.7	0.5
EF	0.45	0.5	0.7	0.5
ET	0.45	0.5	0.7	0.5

The selected WWR is primarily influenced by the proportion of desired or undesired solar heat input, but also partly by the ventilation requirements. In tropical climate zones, good ventilation is essential despite the warm climate, which is why open and airy constructions with larger windows facing each other are favoured. In hot but dry climates, on the other hand, shading and avoiding heat gain are crucial to prevent the interior from overheating. The case studies also show that window size and position vary depending on the climate. The tendency here is as follows: As temperatures rise, there is a tendency to use smaller windows, especially in the south, while cross ventilation (east-west direction) becomes more important. In colder climates, on the other hand, the window area is increased in order to maximise solar heat gain. Effective cross ventilation is also required in humid climates [15], [46].

The exact positioning of certain windows on the façade is difficult to control parametrically. However, it is possible to specify a desired WWR for each compass direction. Based on [47], [48] WWR values are determined for different climate zones per cardinal point. The data were collected and tabulated in Table 5 according to climate zones; if not available, the values were interpolated according to expected climate conditions.

3.2.2 *Solar*

The parameters for the solar analysis include the tilt, the orientation (see Figure 3) and the surface area of the solar panels. The optimum tilt angle and the orientation of the panels are based on the direction from which the sun shines most strongly on average over the course of the year. In order to determine these values, information on the sun's path and the weather conditions is required. Based on this data, solar panels can be oriented so that they receive the maximum amount of sunlight on average. The specified values for inclination, orientation and available surface area can then be used to simulate how much electricity the panels will generate [49], [50].

3.2.3 *Roof*

The selection of the appropriate roof type, flat roof or pitched roof, and the determination of the pitch angle of the pitched roof depend on the occurrence of wind, rain and snow at the desired location. This data can be retrieved from the local weather information and is available for most weather stations as EnergyPlus Weather (EPW) files.

The wind speed is taken directly from the EPW data. Whether an increased wind speed is to be expected is determined as follows: If the highest monthly 15th percentile of the wind speed exceeds 8 m/s (5 Bft [51]), a pitched roof is selected as a suitable choice due to the increased wind load. A similar selection procedure is used for the parameters rain and snow. For rain, a maximum monthly precipitation of 65 mm is the limit chosen for the use of a flat roof. If the value is higher, a pitched roof is recommended. For snowfall, an upper limit of 25 mm per month is used as the criterion for deciding whether a flat roof is still suitable.

4. Methods and tools

The basic functions of the Tiny House Generator should not only be the automatic creation or generation, but also the visualisation and viewing of models. A choice must be made as to the form in which the information is entered, processed and output. There are two approaches for operating the program and displaying the generated models. The first method is to integrate the operation directly within the modelling software in which the building is generated. Such modelling software usually has a view window anyway, in which the geometry of the building is displayed and with which the user can interact. The second method is to integrate the model into a separate program or web application. In this case, the geometry, which was still created within a modelling software, would have to be transferred to the external program to display and edit it there. In order to evaluate the two approaches, a process comparison is carried out on the basis of selected user-friendliness criteria [52].

• Clarity of purpose: How clearly is the function of the system recognisable to the person using it?

• Unity of system design: How coherent is the structure of the system?

• Completeness: Is the system complete in its functions, are all intermediate steps meaningfully listed, and does it offer all necessary tools?

• Error acceptability: How well can the system deal with input errors or incorrect input data on the part of the person using it?

• Training: How much training is required to be able to use the system efficiently?
• User skill: What level of prior knowledge is expected of the user?
• Self-teaching: Is the system designed in such a way that it can be learnt independently?
• Easy access: How easy is it to access the system and its functions?

Table 6. Comparison of procedures

	Generator in modelling software	Generator as an application
Clarity of purpose	0	+
Unity of system design	+	--
Completeness	++	-
Error acceptability	--	0
Training	-	+
User skill	-	0
Self-teaching	--	+
Easy access	-	++
Result	-4	2

A comparison of the different procedures discussed is presented in Table 6. For each criterion, a rating is assigned that is either positive (++ or +), neutral (0), or negative (- or --). At the end, the overall result of each method is determined by adding up the ratings.

In terms of "Clarity of Purpose", a standalone application offers a clear goal for the functions and ensures a targeted operating process. In contrast, although the modelling software also has a clear objective, the complexity of its operation could make it more difficult to focus on the Tiny House Generator. The "Unity of System Design" is almost optimal with an output in the modelling software. The building is generated within the software anyway. The application, on the other hand, requires separate platforms, which makes system integration more complicated.

The generator within the modelling software contains and provides access to all functions and tools used in the generation process. Accordingly, all intermediate steps can also be accessed if required. The application can only display selected results, i.e., the finished model. Although this results in poor "completeness" for the application, it also leads to better "error acceptability". However, this is insufficient for the modelling software. As the user has deep involvement in the process, errors can easily be made when operating the program.

The modelling software requires considerably more instruction and prior knowledge before it can be used effectively, whereas an application can be designed to be much more user-friendly so that it can be operated without any special prior knowledge. Accordingly, it is also difficult to operate the modelling software through self-teaching. The biggest advantage of the application is the "easy access". As web applications are executed in a browser, they are not restricted to a specific operating system and can be used on a variety of devices. In contrast, the generator in the modelling software is more difficult to access, as the software must first be installed and configured on the computer. In addition, a web app offers performance-independent use. This means that the computational intensity of the applications, in particular the complex geometry and energy analyses, can be executed on a powerful server rather than on the user's end device.

Overall, the comparison shows that an application is more user-friendly and accessible, while the modelling software integrates more functions but is more complex and less user-friendly. In view of these advantages, the choice falls in favour of a web application.

4.1 *Modelling software*

The software Rhinoceros 3D Version 7 (Rhino7) is used for the 3D modelling and scripting of the Tiny House Generator. An essential component of the modelling environment used in this work is Grasshopper, a graphical algorithm editor that is closely integrated into Rhino. Grasshopper makes it possible to control modelling processes through visual programming. This is done by using parameters and components that store and process data. The data is represented in the form of nodes that are linked together to create a data flow [53], [54].

Figure 14 shows an example of the use of Grasshopper. In this example, a rectangle with the dimensions 3 m × 2 m is created first. This rectangle is then used in the "Box from rectangle" component to generate a box with a base area of 3 metres × 2 metres and a height of 2 metres. One advantage of Grasshopper is that all intermediate steps remain available, so that you can continue to manipulate the rectangle even after the box has been generated.

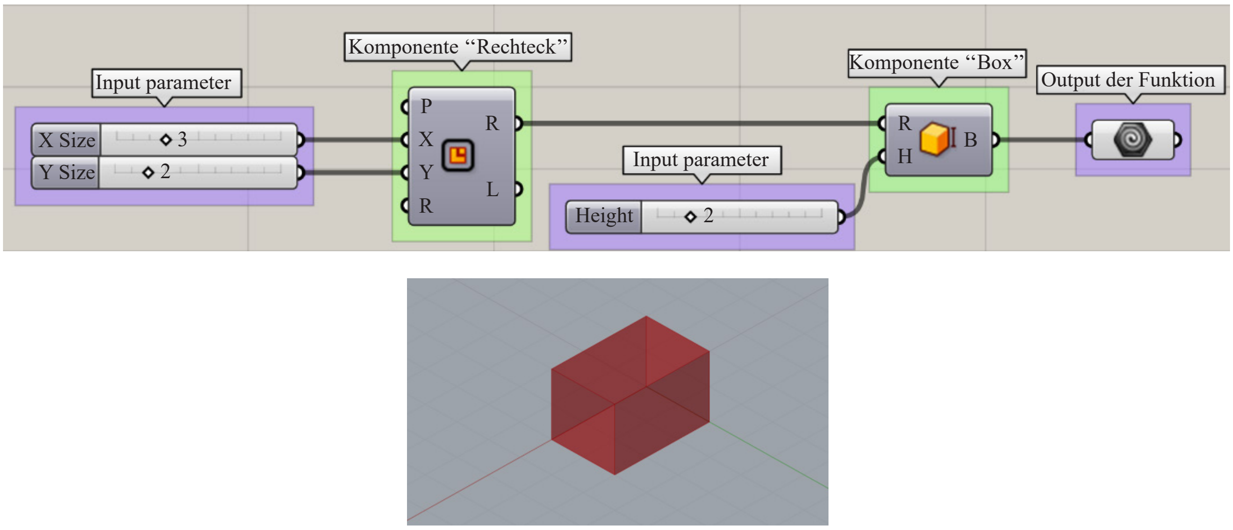

Figure 14. Example of the use of Grasshopper with different components to form a box

Grasshopper offers the option of installing plug-ins that support the energy simulation of buildings. You can also write your own code for the script in the programming languages C# and Python and run the completed scripts on a website via an external interface (see Chapter 5). A web integration makes it possible to control parametric models without direct interaction with the Grasshopper interface, which, as already discussed in Chapter 4, makes operation easier for a complex script such as the Tiny House Generator [54]-[56].

4.2 *Development environment of the web app*

Various providers offering services for the integration of Grasshopper scripts into web-based applications were evaluated for the development and hosting of the Tiny House Generator web application. After thoroughly analysing and contacting several service providers, the decision was made in favour of VIKTOR. The platform handles the data transfer between the web application and the Grasshopper scripts by receiving the user's input in the web interface, transmitting it to the corresponding Grasshopper script and sending the results back to the web application. VIKTOR is characterised by its close integration with Rhino Compute, which enables support for a wide range of Grasshopper plug-ins (see Chapter 5.4). Another decisive factor in the choice of VIKTOR was the existing GIS functionality within the platform. This makes it possible to integrate geographical data into the modelling process, which is particularly interesting for location-based analyses of tiny house designs. The web app in VIKTOR is programmed in Python, so advanced Python-based functions can also be implemented [57].

4.3 *Cataloguing of EPW files*

Although all possible locations have already been categorised by climate zone according to the Köppen-Geiger classification, it makes sense to use more detailed climate data for some calculations. EPW files are suitable for this purpose. EPW files contain data such as the position of the sun, wind speed and temperature, which can be used for the precise simulation of buildings. The download links of the EPW files can be integrated directly into a Grasshopper script so that the weather information can be retrieved in real time. However, the challenge was to identify and efficiently process the specific download links for all 55,121 weather stations [58]. For this purpose, an existing tool from Mingbo Peng [59] was used and slightly modified to adapt it to the required data. The program code works by web scraping the website, extracting all the ZIP files of the EPW data and putting the required information, such as the geographical coordinates and links, into a structured form (When contacting the admins of the "Onebuilding" project regarding the links to the files, the author was informed that this information would in future be available in tabular form on the website itself). The extracted data is then saved in a Comma-Separated Values (CSV) format. The information is then available as follows (Table 7).

Table 7. CSV data format for EPW files

Name	Lat	Long	URL
DZA_AD_Miliana.604300_TMYx.2004-2018.zip	36.3	2.23	http://climate.onebuilding.org/WMO_Region_1_Africa/DZA_Algeria/AD_Ain_Defla/DZA_AD_Miliana.604300_TMYx.2004-2018.zip
CAN_BC_Whistler-Nesters-Spruce.Grove.Park.711750_TMYx.zip	45.92	-60.65	http://climate.onebuilding.org/WMO_Region_4_North_and_Central_America/CAN_Canada/BC_British_Columbia/CAN_BC_Whistler-Nesters-Spruce.Grove.Park.711750_TMYx.zip

The next step is to find the nearest weather station to a given position. The Haversine formula is used to calculate the distance between two points on the earth's surface [60].

$$d = 2r\sin^{-1}\left(\sqrt{\sin^2\left(\frac{\phi_2-\phi_1}{2}\right)+\cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\psi_2-\psi_1}{2}\right)}\right)$$

With d = distance between two points with longitude and latitude (ψ, ϕ), and r = radius of the earth.

The desired position of the building is used as the input value. The distance to all stations is then calculated. Finally, you can filter for the station with the shortest distance to the position.

5. Documentation of the tiny house generator

This chapter contains the documentation of the developed web app. Firstly, the operation of the Tiny House Generator is explained using a practical example. This is followed by the concept documentation of the web application and the Grasshopper script. The implemented solutions are discussed in detail at a technical level.

5.1 *Concept and structure of the web app*

The web app described here acts as a user-friendly front end for the underlying Grasshopper script, which performs the actual analysis and modelling. In the first processing step, the web app enables users to place a point on an interactive map. The local climate zone is determined using an overlaid GeoJSON dataset. This climate information serves as important input for the subsequent 3D modelling, which takes place in the second step, in which the user enters basic building data such as room height and entrance direction. This previously collected data is transferred to the Grasshopper script at the touch of a button, which analyses the building parameters and outputs them as a 3D model. In addition, a 2D view of the floor plan is provided in the second tab. Finally, the results of the analyses are displayed in tabular form in the third step.

5.2 *Practical example: User guidance*

The following is a step-by-step guide to using the Tiny House Generator web application (For a demonstration of the operation and functionality of the Tiny House Generator, please refer to the following video uploaded to the website Youtube: Tiny House Generator Demonstration. The video demonstrates the entire process from site selection to the output of the building model).

5.2.1 *Step 1: Selecting a location using the map*

When opening the website for the first time, the user is presented with a page showing a world map and some input options on the left-hand side. The map has an overlay with Köppen-Geiger climate zones. The display of this overlay can be customised using the "Styling" option, where the opacity and line thickness of the overlay can be changed. A legend explaining the colour assignment to the climate zones can be found in the lower left area of the map.

At the beginning, a red pin is placed on the map by default, representing the location of the planned Tiny House. This pin is initially set to Darmstadt (Germany). To select a different location, the user can either enter the coordinates manually in the "Enter a location" field or delete the pin by clicking on the rubbish bin symbol. Once the "Create Point" button has been clicked, the user can set a new location by clicking on the map.

The climate zone of the last pin set is displayed on the right-hand side of the website. As soon as the desired location has been set, the user can either click on "Next Step" or switch to "Step 2 Building" in the upper band to proceed to the next step.

5.2.2 *Step 2: Generation of the building model*

In the second step, the user is shown an empty view window and fields on the left-hand side where the room height and the azimuth direction of the building entrance can be entered. The azimuth refers to the angle in degrees clockwise from north, where 0° corresponds to north, 90° to east, and so on. Once these parameters have been entered, the simulation can be started by pressing the "Start simulation" button at the bottom right-hand side of the screen. The simulation can take up to four minutes, with the average time required being between 30 and 120 seconds. Once the simulation is complete, the geometry is displayed in the view window. If the simulation fails, the user should adjust the parameters slightly and start a new simulation. Incorrect output or geometry can also occur, as Grasshopper is not always one hundred per cent deterministic. Even then, the building should be regenerated.

In the second tab of step 2, "Floor plan and sections", views of the building structure are presented. The representations of the floor plan and sections should be updated after each new simulation. The "Load current floor plan" button is used for this purpose.

5.2.3 *Step 3: Analysing the building data*

Once the geometry of the simulation has been viewed, the user can proceed to the data analysis by selecting the "Next Step" tab or by clicking on "Step 3 Data" in the upper band. If the user wishes to return to a previous step, this can be done at any time by clicking on the corresponding step in the band or by pressing the "Previous Step" button.

In the third step, the data is displayed in tabular form, supplemented by explanations of this data. The precipitation values for the selected location can also be viewed in the "Weather data" tab.

5.3 *Detailed description of the web app*

For a better overview, the script has been divided into different modules. The modules are the definitions of the user interface (*Parametrisation.py*), the processing of geographical information (*gis_functions.py*), and the control of the app, including the application of the GH script (*app.py*). These modules can be viewed in the Appendix.

5.3.1 *User interface and parameter definition*

The *Parametrisation.py* module defines the user interface and the associated steps. It defines the three steps and initialises the associated parameters:

1. Location selection and climate zone integration: The user can select a location on a map. A default location (Darmstadt) is specified, which is defined in the GeoPointField as DEFAULT_LOCATION. The user can customise the style parameters of the GIS display, such as the opacity and line width of the map display.
2. Definition of building parameters: The user can define geometric parameters for the Tiny House, such as room height and entrance orientation. Two tabs are provided in the user interface: the 3D model and the 2D model area.
3. Data analysis and result evaluation: In the third step, the results of the analysis are presented. The user has no input options; the results are displayed in two tables.

5.3.2 *Geographical information processing*

The *gis_functions.py* module processes geographical data, particularly the determination of climate zones based on the location coordinates. First, climate zones are assigned to specific colours that are stored in a dictionary. The get_gdf() function creates a GeoDataFrame from a geo-JSON file in which the climate zones are stored and applies the style parameters selected by the user. The find_climate_zone() function determines the climate zone of a location based on latitude and longitude. The function Create_legend() creates a legend for the climate zones with the assigned colours so that they can be displayed on the map.

5.3.3 *JSON processing*

The *json_utilities.py* module contains functions for processing JSON data. The read_json_file() function reads JSON files and returns the text lines they contain. This data is further processed by parse_text_lines() and parse_column_lines() to retrieve the relevant parameters from the file.

5.3.4 *Application logic and integration*

The *app.py* module forms the centrepiece of the application and coordinates the various modules. It comprises the familiar steps:

1. GeoJSON and data view: The get_geojson_view function is used to display an interactive map with various geographical information. The user can select a specific point on the map and the application then displays the corresponding climate zone. In addition, the geographical data is visualised on the map in a specific display style. At the beginning of the function, a so-called GeoDataFrame (abbreviated gdf) is retrieved. This GeoDataFrame contains the geographical data to be displayed on the map. To do this, the function get_gdf is called, which returns this data based on the style parameters defined by the user in step_1.styling. The GeoDataFrame is then converted into a GeoJSON format. A MapLabel is created to mark the position on the map. When the user selects a new point on the map, the coordinates of this point are saved in the parameters params.step_1.point.GeoPointField. The function checks whether the coordinates are available and valid. If this is the case, the coordinates (latitude and longitude) are extracted and used to create a new point (point_geojson) on the map and to display this new point. The extracted coordinates are rounded to display them in a standardised format. The longitude is rounded to three decimal places, the latitude to two.

   Then, it calls the function find_climate_zone to determine the climate zone of the selected location based on the coordinates. This function searches the GeoDataFrame for this purpose. A string is created based on the result of the climate zone determination. If a climate zone is found, the message contains the latitude, longitude, and the corresponding climate zone. The function then creates a DataGroup containing the string created above. Finally, the information and the map are passed on for display as GeoJSONAndDataResult.
2. 3D model and floor plan display: At the beginning of the run_grasshopper function, the path to the Grasshopper script responsible for generating the tiny house is defined. The function then extracts all parameters from the previous steps. This includes geographical information, such as latitude and longitude, the room height of the Tiny House and

the orientation of the entrance door (azimuth), as well as the climate zone, which was determined in a previous step. The extracted parameters are transferred to a dictionary (formatted_params). This dictionary is required so that the parameters are available in a format that they can be assigned to the Hops components of the same name in Grasshopper. The Grasshopper script is executed with the formatted parameters.

The results of the Grasshopper analysis are temporarily stored in memory. This enables the data to be loaded and further processed in later steps. The memory checks whether parameters have been changed each time it is loaded and starts a new simulation if changes are made. The model created in Grasshopper is converted into a Rhino 3D format (file3dm). The add_objects_to_model function inserts the geometric objects in JSON format into the 3D model. Finally, the function returns the 3D model, which is used by the application for display in the user interface.

The view_floorplan function is used to display the floor plan and sections of the Tiny House in a 2D view. The function loads the Grasshopper output saved by the previous function from the memory. Similar to the run_grasshopper function, the geometric data representing the floor plan is transferred to a Rhino 3D format (file3dm). The add_objects_to_model function is used again to add the individual elements of the floor plan to the model. Once the 2D geometry objects have been added to the model area, the geometry is saved as a Rhino 3D file and returned for display in 2D.

3. Data analysis and description: Step 3 comprises two functions. The first function, run_data_analysis, is responsible for the tabular representation of the parameters used in the modelling. This function reads the JSON file from the memory using the read_json_file function and processes the data it contains using the parse_column_lines function. The function then creates the data for the table view. Two lists, table_data and row_headers, are initialised. "row_headers" contains the keys of the processed data as row headers, while table_data contains the corresponding values. For each key-value pair in the processed data, the key is added as a row header and the value as table content. The result is finally returned as a TableResult object.

5.4 *Interface between Rhino-compute and hops*

A large part of Rhino's functionality is available via the Software Development Kit (SDK) "RhinoCommon". An SDK is a collection of tools, libraries, and documentation that allows developers to create applications for a specific software platform. RhinoCommon allows applications to be built on top of Rhino by making a variety of Rhino functionalities accessible. For example, it is possible to run Rhino within Windows applications using Rhino.Inside. Applications that use Rhino.Inside run Rhino without a user interface, i.e., "headless". This means that Rhino runs without the usual graphical user interface and without an associated document, and that 3D geometries can only be manipulated using scripts and the RhinoCommon library [61], [62]. A disadvantage of this methodology is that some Grasshopper plug-ins and some custom code are not compatible with Rhino.Inside due to this mode of operation [61].

Rhino Compute is an open-source project based on Rhino.Inside and provides the RhinoCommon SDK as a stateless Representational State Transfer Application Programming Interface (REST API). A REST API makes it possible to address web services via the HyperText Transfer Protocol (HTTP) protocol. This is also stateless, which means that each call to the API is self-contained and no information about previous interactions is saved. Rhino Compute is therefore a kind of web server that can perform geometry calculations using the Rhino geometry library. In addition to providing these basic functions, Rhino Compute also enables the calculation of Grasshopper definitions [61], [62].

For the Tiny House Generator application, a Rhino Compute server is run on a local computer so that it is possible to solve Grasshopper definitions from the web and return the result. Hops represents an interface to Rhino Compute within a Grasshopper function. The Hops client passes the Grasshopper definition to an instance of the Rhino and Grasshopper server, which is known as the "Rhino Worker". In this context, the Rhino Worker is an instance of Rhino Compute. The Rhino Worker solves the Grasshopper definition sent by the Hops client and then returns the result [56].

In the Tiny House Generator application, the inputs for the Hops component are processed via the Viktor for Grasshopper interface. This interface sends input through the Hops components of the Grasshopper function to the Rhino Worker, which returns the resulting data and geometries. These are then sent back to the Viktor for Grasshopper interface to be displayed on the website [56], [57].

5.5 *Concept and functionality of the Grasshopper script*

The underlying Grasshopper script performs the complete analysis and modelling of the Tiny House based on the data entered by the user in the web app (Table 8). Firstly, the script collects all the relevant climate data required to calculate the building parameters. This data is used to calculate factors such as the optimum insulation thickness. In the next step, the script generates the floor plan of the building, followed by the placement of the building openings. The walls, foundations, and roof are then modelled. A solar simulation is carried out to determine the potential for photovoltaic systems. Finally, the calculated data is processed, sorted, and prepared for transfer to the web app so that it can display the 3D visualisation and analysis results.

Table 8. Inputs and outputs of the Grasshopper script

Inputs	Outputs
Room height in metres	Geometry as mesh
Latitude and longitude of the building	Data on the building
Direction of the entrance door	
Climate zone of the building	

5.6 *Plug-ins used*

Various specialised plug-ins were integrated into Grasshopper for the Tiny House Generator function, each of which provides specific functions. Care was taken to ensure that the plug-ins are compatible with Rhino Compute:

• Pollination Suite (mainly Ladybug): This plug-in is mainly used for climate and solar analyses. Ladybug enables a detailed analysis of the sun paths and climatic conditions at the location of the building [55], [63].

• DeCoding Spaces + Magnetising FPG: These plug-ins are used to create a basic floor plan. DeCoding Spaces contains generative components for algorithmic architectural and urban planning; together with Magnetising FPG (Floor Plan Generator) these can be used to place and adapt floor plans in a predefined space.

• Sasquatch: Sasquatch is used for geometry manipulation and offers specific tools that are important for performance in this script. For example, Sasquatch enables faster, multi-core calculation of geometry interactions.

• PlanFinder: This plug-in supports the furnishing and checking of the logic of the generated floor plan. PlanFinder uses a library of as-built plans that are compared with the specified form (PlanFinder requires a licence. However, a 30-day trial version can be used).

5.6.1 *Description of the Grasshopper script*

This section documents the Grasshopper script, which is divided into several modules, each of which fulfils specific tasks and builds on one another. These modules are in turn subdivided into submodules. These are named in alphabetical order. The order roughly represents the technically logical sequence of operations. The complete script is shown in the appendix with the submodules labelled. A flowchart is found in the Appendix.

Module 1 forms the foundation of the geometry generation (Table 9). It begins with defining key parameters such as the building footprint, floor heights, and openings. Based on these inputs, the system creates the primary geometries for the house, including exterior and interior walls and furniture. The focus lies on organizing the structural building elements according to the user-defined inputs.

Table 9. Module 1: Information collection and processing

Module 1	Information collection and processing	
Inputs	Outputs	Task
Room height in metres. Latitude of the building. Longitude of the building. Direction of the entrance door. Climate zone of the building. CSV files (climate information).	Source of the climate data. Design features (construction method, insulation thickness, etc.). Floor plan with furnishings. Polyline that marks the inner edge of the outer wall.	Collect and process input data. Consolidate and link the data for further use in the script. Provision of climate data, design features, and floor plan information for the subsequent modules.

Module 2 handles the generation of floor slabs and openings in the building (Table 10). The building outline is analyzed to generate suitable geometries.

Table10. Module 2: Placement of openings and interior walls

Module 2	Placement of openings and interior walls	
Inputs	Outputs	Task
Polyline marking the inner edge of the outer wall. Position of the interior walls. Position of the entrance door. WWR for each sky direction. Thickness of the outer walls.	Geometries for windows, doors, and interior walls. Geometric information about the exact positions of these elements.	Placement of windows, doors, and interior walls based on the defined input data.

Module 3 focuses on generating the outer wall structures (Table 11). Walls are modeled as layered systems, including load-bearing cores, insulation, and interior finishes. The connections between walls and slabs are further detailed with elements like ring beams. The module distinguishes between solid and timber construction methods, applying geometry logic specific to each type.

Table 11. Module 3: Generation of the exterior walls

Module 3	Generation of the outer walls	
Inputs	Outputs	Task
Construction features (construction method, insulation thickness, roof shape). Polyline that marks the inner edge of the outer wall. Position of wall openings.	Geometry of the exterior walls. Edge curve of the building.	Generate the external walls based on the defined parameters. Integration of openings (windows, doors) into the exterior walls. Adaptation of the façade to the defined roof shape.

Module 4 is dedicated to roof generation and the integration of photovoltaic systems (Table 12). Depending on the climate, either flat or pitched roofs are generated. Flat roofs consist of layered structures like sloped screed, insulation, and waterproofing. Pitched roofs are constructed with a more complex logic, taking roof pitch and orientation into account. Roof elements such as rafters, counter battens, and the final roof cladding are generated accordingly. Simultaneously, a solar analysis is performed: climate data is loaded, solar radiation on the roof surfaces is simulated, and optimal positions for photovoltaic panels are determined. The panels are then placed, oriented, and aligned based on yield optimization, and the potential energy output is calculated.

Table 12. Module 4: Generation of the roof areas

Module 4	Generation of the roof surfaces	
Inputs	Outputs	Task
Edge curve of the building. Design features (e.g., roof pitch, roof overhang). Source of the climate data.	Geometry of the roof surfaces. Geometry of the solar panels. Simulated yield of the solar panels.	Generation of the roof surfaces based on the specified parameters and design features. Optimisation of solar panel placement according to weather data and location. Simulation of the energy yield of the solar system for the specific roof construction and site conditions.

Module 5 manages the final processing of all generated data and geometries for export to the web application (Table 13). This includes collecting and converting all 3D geometries, generating building sections through intersection planes, and converting objects into colored meshes for better visualization. Additionally, all 2D drawings and textual information are compiled. The goal is to prepare and deliver all relevant data in a clean, structured format for web-based display and interaction.

Table 13. Module 5: Data processing for output

Module 5	Data processing for output	
Inputs	Outputs	Task
Geometries of all objects (e.g., walls, roof surfaces, windows, doors, solar panels).	Edited geometries and data for export to the web app.	Collect all geometries. Create cuts. Conversion and preparation of the generated geometries for handover to the web application. Ensure the compatibility of the geometry data with the web app interface.

6. Examples of generated tiny houses

To demonstrate the Tiny House Generator, buildings were designed at three different locations—Darmstadt (Cfb), Phoenix (BWh), and Oulu (Dfc). The geometry was taken from the generator and slightly modified in each case to show how the generated buildings can be further developed through minor adjustments. For example, the furnishings, which are only displayed as a floor plan in the Generator, were manually revised and modeled in 3D. The three buildings are presented below. Views, floor plans, and further renderings can be found in the Appendix.

6.1 *Tiny house for Darmstadt*

The Tiny House in Darmstadt was customised for the local climatic conditions of the Cfb climate zone (Figure 15). The monolithic façade has a ventilated insulation system that helps to remove the moisture that forms in the façade during winter. The load-bearing structure is made of concrete, which reduces the day-night temperature fluctuations typical in southern Hesse, owing to its higher thermal mass. The insulation thickness of the external façade was set at 12 cm in accordance with the LCCA. The floor area of the building is 35 m^2, which is compact but still generous for a tiny house. The roof was designed as a pitched roof in order to fulfil the requirements of the structural conditions with regard to rain and snow loads.

Figure 15. Rendering of the Cfb building—Interior

The solar collectors are oriented at an inclination of 28.83° and an azimuth of 186.88°, achieving an average energy yield of 7.49 kWh per day. The floor plan of the house features an open-plan layout that combines the living/sleeping area and kitchen. The bathroom is housed in a separate room.

On the south side, there is a floor-to-ceiling window that leads to an integrated terrace with a pergola, which can be fitted with awnings to provide protection from the sun. This creates a visual extension of the interior into the outside and brings additional daylight into the living area (Figure 16).

Figure 16. Rendering of the Cfb building—(a): outside from the south; (b): rendering outside from the west

Compared to the original generator output, the model was adjusted so that the building opening on the south side was designed as a door instead of a standard window. In addition, the terrace area was added manually.

6.2 *Tiny house for Phoenix*

The Tiny House in Phoenix, Arizona, is specially adapted to the conditions of climate zone BWh. Due to the extreme heat and dryness, the location requires different structural adaptations. The building is characterised by a compact, cubic shape with a smaller south-facing side to minimise heating from direct sunlight. A key component of the design is the reduction of window areas on the south side. Only a narrow strip of glazing was used there, which continues to a larger window on the west side. This provides the interior with sufficient daylight without gaining excessive heat (Figure 17).

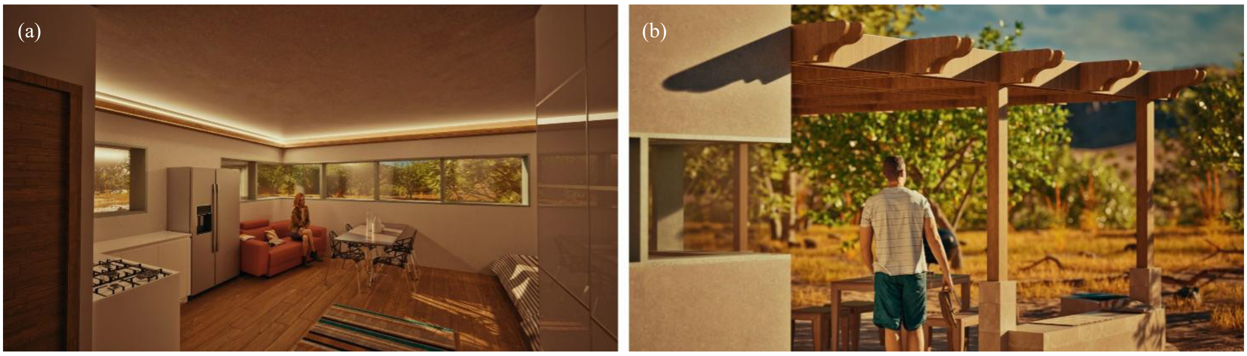

Figure 17. Rendering of building Bwh—(a): inside from the entrance; (b): outside to the south

A pergola on the south side serves as additional sun protection and also creates a shady outdoor area for the residents. This measure reduces the direct heating of the façade and allows users to stay in the shade despite the heat outside. The building was also painted in light colours to minimise the absorption of solar radiation and thus reduce heat gain. The façade consists of monolithic concrete walls with 9 cm of insulation. The flat roof is an efficient solution for this region, as it absorbs less direct sunlight and there is no structural need for a pitched roof. With a floor area of 37 m^2, the Tiny House offers enough space for two people while utilising the space efficiently. The solar panels on the flat roof are oriented with an inclination of 27.64° and an azimuth of 192.46°. This configuration provides an average of 2 kWh of energy per day (Figure 18).

Figure 18. Rendering building Bwh—(a): rendering outside from the east; (b): rendering outside from the north

Compared to the original design, the window bands on the south side have been enlarged. In addition, the pergola has been added.

6.3 *Tiny house for Oulu*

The third Tiny House was designed for Oulu, Finland, in the Dfc climate zone. The building stands out particularly in comparison to the one in Arizona due to its elongated shape. The large south-facing side offers as much solar heat as possible. The architecture relies on warm materials and colours that create an inviting, homely atmosphere. The large windows on the south-facing side allow good use of daylight in the living area and support passive solar heating. The adjoining, spacious terrace creates a pleasant outdoor area that is easy to use in the warmer months. The floor area of the building is only 32.14 m^2, but by integrating a sleeping loft in the form of a gallery with its own window in the gable, enough space is created to accommodate an office in the lower area (Figure 19).

Figure 19. Rendering building Cfb—(a): rendering outside from the south; (b): inside

A timber frame construction is used as the supporting structure. The insulation in this project is more extensive than in the other designs. There is a 20 cm thick layer of insulation between the posts of the timber frame construction, which is supplemented by a second 10 cm thick layer of insulation. In order to cope with the high snow loads in this region, a pitched roof was chosen. This is equipped with solar panels, which are aligned to an inclination of 31.03° and an azimuth of 186.70°, resulting in an average energy yield of 4.55 kWh per day (Figure 20).

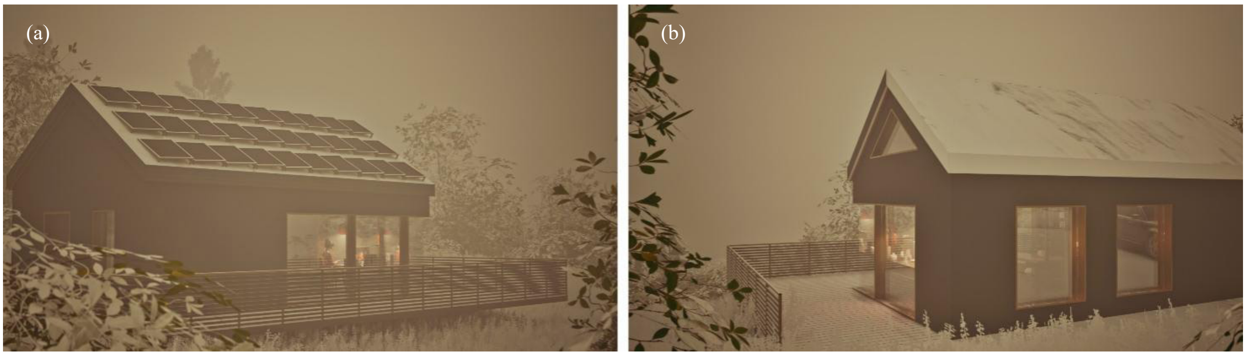

Figure 20. Rendering of the Cfb building—(a): rendering from the south-west; (b): rendering from the north-east

Here, the basic structure of the building was changed compared to the generator output in that the windows were connected across corners and made floor-to-ceiling, and the loft mentioned above was modeled by hand.

7. Evaluation

This research has successfully completed the development of a user-friendly tiny house generator based on fully parametric modelling using Grasshopper. The script allows a variety of customisations to be made and various design parameters for tiny houses to be modified. A major success was the integration of different Grasshopper tools together and to extend the functionality through custom programming. The script creates a basis that allows for future extensions due to its parametric nature. For example, it would be possible to integrate additional functions such as the dimensioning of technical building equipment (e.g., heating and cooling requirements). Another interesting extension would be the generation of larger buildings, as the current generator is limited to single-storey buildings. The implementation of a "Tiny House on Wheels" option, which would have to use lighter materials such as wood due to the weight requirements, would also be an exciting challenge. The integration of structural analyses, for example using the Karamba3D plug-in for finite element analyses, could also significantly increase the level of detail of the structure. This would be a useful addition in addressing extreme climates with extremely heavy wind forces or snow loads. As it currently stands, the tool only generates within a predefined range between maximum and minimum of certain aspects.

By cataloguing the EPW files and developing an interface between Grasshopper and the OneBuilding server using a Python script, it has been possible to develop a tool that automatically identifies the nearest weather station for a selected location. This tool is not only an important component for the Tiny House Generator but also provides a valuable tool for other parametric design projects by drastically simplifying access to accurate climate data for the analysis and optimisation of buildings.

The methodology of climate-related design pursued in this work offers an approach that considers the interplay between climate and building comfort. Nevertheless, a central problem remains: Planning is inevitably based on historical climate data ("building for tomorrow with yesterday's data") [30]. This limits the adaptability of buildings designed according to this methodology. While the generator is able to react fundamentally to climatic and constructional conditions, one aspect of architectural individuality remains unconsidered. Generated houses tend to appear *generic*, as algorithmic systems leave little room for aesthetic or cultural particularities. However, architecture is strongly characterised by individual preferences, local building traditions and aesthetic values, which are difficult to capture in a parametric system. Although the generator provides well-founded suggestions for building principles, it can hardly offer a subjectively appealing solution. Local laws, natural risks such as earthquakes or specific regional challenges are further factors that are so complex that they are difficult to integrate into an algorithm. One possible solution to these challenges could be the use of Artificial Intelligence (AI). By training neural networks based on large case studies, a system could be developed that is able to better take into account local characteristics and complex relationships between climate, construction, culture and geographical location. However, such an approach would involve a considerable amount of training and would first require a methodology to define relevant parameters and subsequently extract them from building plans, as-built surveys or other data sources.

Another problem identified in the work concerns the performance of the system. Although a web app was developed due to these restrictions, the calculation speed is limited by the computing power of the host. It would be possible to improve performance by using more powerful servers in order to shorten computing times and provide users with a smooth application. Overall, the work provides a basis for the future development of a parametric system for generating buildings. For the future, the question remains as to how the generator can be further optimised to increase its functionality and accuracy. Currently, many calculations are based on standardised values that are common in the industry, for example for the roof pitch. It would be a great advantage to be able to use more specific verifications, for example to determine the optimum roof angle more precisely depending on wind, snow and rain. Although weather and climate data are readily available, there is a lack of accessible detailed analyses that go beyond normative specifications.

In conclusion, it can be said that the Tiny House Generator developed in this research already represents a significant step towards automated construction planning. Compared to other tools, the project presented here has the advantage of being free to use and, since Grasshopper and the project's source code have been made available online, it can also be adapted by anyone. In addition, the output file can be manipulated by most common 3D Computer-Aided Design (CAD) programs. With further open-source development, it could become an even more powerful and flexible tool that can be used in both architecture and construction. Unlike AI-based tools, this approach offers full transparency: every step of the process can be clearly understood, followed, and modified from start to finish. This makes the method not only reliable but also accessible for personal adaptation. At the same time, it must be acknowledged that artificial intelligence could potentially offer advantages in terms of flexibility, as it will adapt better to unknown environments or unforeseen requirements. Nevertheless, the workflows and insights shown here could themselves serve as a foundation or even training material for future AI systems.

Conflict of interest

The authors declare no competing financial interest.

References

[1] Bürgerservice Hessenrecht, ‘‘Hessisches wohnungsaufsichtsgesetz: HWoAufG [Hessian housing supervision act: HWoAufG],’’ *Bürgerservice Hessenrecht*, 1974. [Online]. Available: https://www.rv.hessenrecht.hessen.de/perma?d=jlr-WoAufGHErahmen. [Accessed August 31, 2025].

[2] A. Morrison and G. Morrison, *Tiny House Designing, Building, & Living*. Indianapolis, IN: Penguin, 2017.

[3] R. Mitchell, *Tiny Houses Built with Recycled Materials: Inspiration for Constructing Tiny Homes Using Salvaged and Reclaimed Supplies*. Avon, Massachusetts: Adams Media, 2016.

[4] M. Sarkar, ‘‘How American homes vary by the year there were built,’’ *U.S. Census Bureau*, 2024. [Online]. Available: https://www.census.gov/content/dam/Census/programs-surveys/ahs/working-papers/Housing-by-Year-Built.pdf. [Accessed May 3, 2024].

[5] U.S. Census Bureau, Current Population Survey, ‘‘Historical households tables: Table HH-4. Households by size: 1960 to present,’’ *U.S. Census Bureau*, 2024. [Online]. Available: https://www.census.gov/data/tables/time-series/demo/families/households.html. [Accessed May 3, 2024].

[6] K. Rechsteiner, *Tiny House: Das Grosse Praxisbuch: Planung, Selbstbau und Fertighäuser [Tiny House: The Big Practical Guide: Planning, Self-Build and Prefabricated Houses]*. Aarau, München: AT Verlag, 2020.

[7] L. Baker, *XS-Small Houses Big Time*, 1st ed., Salenstein, Switzerland: Braun, 2017.

[8] L. Walker, *Kleine Häuser: Geschichte, Skizzen, Baupläne [Small Houses: History, Sketches, Construction Plans]*. Köln: Könemann, 2000.

[9] Japan Ministry of Internal Affairs and Communications, ‘‘Housing and land survey/2018, tatami units of dwelling rooms per dwelling: Table number 13,’’ *Statistics Bureau of Japan*, 2018. [Online]. Available: http://www.stat.go.jp/english/data/jyutaku/index.htm. [Accessed May 3, 2024].

[10] A. Mutter, “Growing tiny houses: Motivations and opportunities for expansion through niche markets,” M. S. thesis, Lund University, Sweden, 2013.

[11] M. Meijs and U. Knaack, *Bauteile und Verbindungen [Components and Connections]*. Berlin: De Gruyter, 2009.

[12] B. Ziegler, *Zum Einfluss von Bauteilintegrierten Latentwärmespeichern auf das Sommerliche Thermische Verhalten von Innenräumen in Deutschland [On the Influence of Building-Integrated Latent Heat Storage on the Summer Thermal Behavior of Indoor Spaces in Germany]*, 1st ed., Aachen: Shaker Verlag, 2016.

[13] A. Reilly and O. Kinnane, “The impact of thermal mass on building energy consumption,” *Applied Energy*, vol. 198, pp. 108-121, 2017.

[14] T. Kuczyński, A. Staszczuk, M. Gortych, and R. Stryjski, “Effect of thermal mass, night ventilation and window shading on summer thermal comfort of buildings in a temperate climate,” *Building and Environment*, vol. 204, pp. 108126, 2021.

[15] G. Lippsmeier, *Tropenbau: Building in the Tropics*, 2nd ed., München, Germany: Callwey, 1980.

[16] C. Schittich, *Im Detail: Mikroarchitektur: Kleine Bauten, Temporäre Strukturen, Raumzellen [In Detail: Microarchitecture: Small Buildings, Temporary Structures, Spatial Cells]*. Germany: Detail, 2010.

[17] U. Knaack and E. Koenders, *Bauphysik der Fassade: Prinzipien der Konstruktion [Building Physics of the Facade: Principles of Construction]*. Basel, Switzerland: De Gruyter, 2018.

[18] G. N. Tiwari, A. Tiwari, and S. M. Srinivas, *Handbook of Solar Energy: Theory, Analysis and Applications*. Singapore: Springer, 2016.

[19] *Handbuch Geneigte Dächer [Handbook for Pitched Roofs]*. Braas GmbH, 2019.

[20] A. Pech, W. Hubner, and F. Zach, *Flachdach [Flat Roof]*. Basel, Switzerland: Birkhäuser, 2021.

[21] K. Verichev, M. Zamorano, C. Salazar-Concha, and M. Carpio, “Analysis of climate-oriented researches in building,” *Applied Sciences*, vol. 11, no. 7, pp. 3251, 2021.

[22] T. Michael, *Diercke Weltatlas [Diercke World Atlas]*, 1st ed., Braunschweig, Germany: Westermann, 2015.

[23] M. C. Peel, B. L. Finlayson, and T. A. McMahon, “Updated world map of the Köppen-Geiger climate classification,” *Hydrology and Earth System Sciences*, vol. 11, no. 5, pp. 1633-1644, 2007.

[24] C. Beck, J. Grieser, M. Kottek, F. Rubel, and B. Rudolf, “Characterizing global climate change by means of Köppen climate classification,” Deutscher Wetterdienst, Offenbach, Germany, Report Number: Klimastatusbericht

2005, 139-149, 2005.
[25] H. E. Beck, T. R. McVicar, N. Vergopolan, A. Berg, N. J. Lutsko, A. Dufour, Z. Z. Zeng, X. Jiang, A. I. J. M. Van Dijk, and D. G. Miralles, “High-resolution (1 km) Köppen-Geiger maps for 1901-2099 based on constrained CMIP6 projections,” *Scientific Data*, vol. 10, no. 1, pp. 724, 2023.
[26] E. Ne’Eman, “Visual aspects of sunlight in buildings,” *Lighting Research & Technology*, vol. 6, no. 3, pp. 159-164, 1974.
[27] M. Boubekri, *Daylighting, Architecture and Health: Building Design Strategies*, 1st ed., Amsterdam, Boston, London: Architectural Press, 2008.
[28] E. T. Ochedi and A. Taki, “Energy efficient building design in Nigeria: An assessment of the effect of the sun on energy consumption in residential buildings,” *Journal of Energy and Architecture*, vol. 7, no. 1, pp. 51-61, 2019.
[29] R. V. Ralegaonkar and R. Gupta, “Design development of a static sunshade using small scale modeling technique,” *Renewable Energy*, vol. 30, no. 6, pp. 867-880, 2005.
[30] J. Graser, A. Staufer, and C. Meier, *Architektur Klima Atlas: Klimabewusst entwerfen in Geschichte, Forschung, Lehre und Praxis [Architecture Climate Atlas: Climate-Conscious Design in History, Research, Teaching and Practice]*. Zürich, Switzerland: Park Books, 2024.
[31] K. Verichev and M. Carpio, “Climatic zoning for building construction in a temperate climate of Chile,” *Sustainable Cities and Society*, vol. 40, pp. 352-364, 2018.
[32] J. Kwiatkowski and B. Arndt, *Basic: Eine Einführung in 10 Lektionen mit Zahlreichen Programmbeispielen, 95 Übungsaufgaben und Deren Vollständigen Lösungen [Basic: An Introduction in 10 Lessons with Numerous Program Examples, 95 Exercises and Their Complete Solutions]*. Berlin, Heidelberg: Springer, 1983.
[33] J. Kwiatkowski and B. Arndt, “Ein- und ausgabeanweisungen [Input and output instructions],” in *Basic. Informationstechnik und Datenverarbeitung [Basic. Information Technology and Data Processing]*, Berlin, Heidelberg: Springer, 1983, pp. 35-46.
[34] F. Bocchini, “Form statt technik: Untersuchung zur ersetzung der haustechnik durch architektonische maßnahmen [Form instead of technology: An investigation into the replacement of building services by architectural measures],” M. S. thesis, Graz University of Technology, German, 2016.
[35] J. Heistermann, *Genetische Algorithmen: Theorie und Praxis Evolutionärer Optimierung [Genetic Algorithms: Theory and Practice of Evolutionary Optimization]*. Wiesbaden, Germany: Vieweg+Teubner Verlag, 1994.
[36] A. Gülten and U. Teoman Aksoy, “Optimum insulation thickness with respect to heating and cooling degree days for different geographical regions of Turkey,” *Energy and Buildings*, vol. 45, pp. 45-50, 2012.
[37] K. Verichev, A. Serrano-Jiménez, M. Carpio, Á. Barrios-Padura, and C. Díaz-López, “Influence of degree days calculation methods on the optimum thermal insulation thickness in life-cycle cost analysis for building envelopes in Mediterranean and Semi-Arid climates,” *Journal of Building Engineering*, vol. 79, pp. 107783, 2023.
[38] M. Abascal, *Tiny Homes: Maximum Style*. Barcelona, Spain: Loft, 2017.
[39] L. Baker, *Tiny Living Spaces: Innovative Design Solutions*, 1st ed., Salenstein, Switzerland: Braun, 2023.
[40] Arup, *Building Energy Efficiency Guideline for Nigeria.* Nigeria: Federal Ministry of Power, Works and Housing, 2016.
[41] UPM Global, “Hinterlüftung von fassaden—ein muss [Rear ventilation of facades—a must],” *UPM Global*, 2009. [Online]. Available: https://bau.net/forum/aussenwaende/15149-1370978425-1.pdf. [Accessed August 31, 2025].
[42] V. Olshevskyi, E. Statsenko, T. Musorina, D. Nemova, and A. Ostrovaia, “Moisture transfer in ventilated facade structures,” *MATEC Web of Conferences*, vol. 53, pp. 1010, 2016.
[43] O. Kaynakli, “A review of the economical and optimum thermal insulation thickness for building applications,” *Renewable and Sustainable Energy Reviews*, vol. 16, no. 1, pp. 415-425, 2012.
[44] Y. Huang and J. Niu, “Optimal building envelope design based on simulated performance: History, current status and new potentials,” *Energy and Buildings*, vol. 117, pp. 387-398, 2016.
[45] S. Park, J. Shim, and D. Song, “Issues in calculation of balance-point temperatures for heating degree-days for the development of building-energy policy,” *Renewable and Sustainable Energy Reviews*, vol. 135, pp. 110211, 2021.
[46] L. Lefaivre, A. Tzonis, and B. Stagno, *Tropical Architecture: Critical Regionalism in the Age of Globalization*. Chichester, UK: Wiley-Academy, 2001.
[47] F. Goia, “Search for the optimal window-to-wall ratio in office buildings in different European climates and the implications on total energy saving potential,” *Solar Energy*, vol. 132, pp. 467-492, 2016.
[48] S. Sayadi, A. Hayati, and M. Salmanzadeh, “Optimization of window-to-wall ratio for buildings located in different climates: An IDA-indoor climate and energy simulation study,” *Energies*, vol. 14, no. 7, pp. 1974, 2021.
[49] H. Moghadam, F. F. Tabrizi, and A. Z. Sharak, “Optimization of solar flat collector inclination,” *Desalination*, vol.

265, no. 1-3, pp. 107-111, 2011.
[50] E. D. Mehleri, P. L. Zervas, H. Sarimveis, J. A. Palyvos, and N. C. Markatos, “Determination of the optimal tilt angle and orientation for solar photovoltaic arrays,” *Renewable Energy*, vol. 35, no. 11, pp. 2468-2475, 2010.
[51] E. L. Delmar-Morgan, “The beaufort scale,” *Journal of Navigation*, vol. 12, no. 1, pp. 100-102, 1959.
[52] K. Meyer and H. Mike, “User friendliness,” *MIS Quarterly*, vol. 8, pp. 1-3, 1984.
[53] Scott Davidson, “About grasshopper,” *Grasshopper*, 2024. [Online]. Available: https://www.grasshopper3d.com/. [Accessed Aug. 12, 2024].
[54] L. Yang, X. Liu, and F. Qian, “Optimal configurations of high-rise buildings to maximize solar energy generation efficiency of building-integrated photovoltaic systems,” *Indoor and Built Environment*, vol. 28, no. 8, pp. 1104-1125, 2019.
[55] M. Sadeghipour Roudsari, M. Pak, and A. Viola, “Ladybug: A parametric environmental plugin for grasshopper to help designers create an environmentally-conscious design,” in Proceedings of Building Simulation 2013: 13th Conference of IBPSA, 2013.
[56] A. Payne, “What is hops,” *RhinoDeveloper*, 2022. [Online]. Available: https://developer.rhino3d.com/guides/compute/what-is-hops/. [Accessed Aug. 7, 2024].
[57] VIKTOR, “Rhino/grasshopper,” *VIKTOR*, 2024. [Online]. Available: https://docs.viktor.ai/docs/create-apps/software-integrations/rhino-grasshopper/. [Accessed Aug. 7, 2024].
[58] Lawrie, K. Linda, and D. B. Crawley, “Development of global typical meteorological years (TMYx),” *Climate. OneBuilding*, 2022. [Online]. Available: https://climate.onebuilding.org/. [Accessed August 31, 2025].
[59] M. Peng, “GetOneBuildingEPW,” *GitHub*, 2025. [Online]. Available: https://github.com/MingboPeng/GetOneBuildingEPW. [Accessed August 31, 2025].
[60] N. R. Chopde and M. K. Nichat, “Landmark based shortest path detection by using A* algorithm and Haversine formula,” *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 1, no. 2, pp. 298-302, 2013.
[61] ShapeDiver Help Center, “ShapeDiver <-> Rhino compute,” *ShapeDiver Help Center*, 2024. [Online]. Available: https://help.shapediver.com/doc/shapediver-rhino-compute. [Accessed Aug. 7, 2024].
[62] B. Gillespie, “Compute: Features,” *RhinoDeveloper*, 2020. [Online]. Available: https://developer.rhino3d.com/guides/compute/features/. [Accessed Aug. 7, 2024].
[63] E. Naboni, A. Maccarini, I. Korolija, and Y. Zhang, “Comparison of conventional, parametric and evolutionary optimization approaches for the architectural design of nearly zero energy buildings,” in Proceedings of Building Simulation 2013: 13th Conference of International Building Performance Simulation Association (IBPSA), 2013.

Appendix: Flow of data in the Grasshopper script

Grasshopper skript complete