ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

A MULTI LAYER CELLULAR AUTOMATA TO PARTICIPATORY ARCHITECTURE: POST-DISASTER CONTAINER SETTLEMENTS

M.Sc. THESIS

Hatice Seray ÖZBAY

Department of Informatics

Architectural Design Computing Programme

SEPTEMBER 2025

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

A MULTI LAYER CELLULAR AUTOMATA TO PARTICIPATORY ARCHITECTURE: POST-DISASTER CONTAINER SETTLEMENTS

M.Sc. THESIS

Hatice Seray ÖZBAY
(523221006)

Department of Informatics

Architectural Design Computing Programme

Thesis Advisor: Assoc. Prof. Dr. Sema ALAÇAM

SEPTEMBER 2025

ISTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

KATILIMCI MİMARLIKTA ÇOK KATMANLI HÜCRESEL ÖZDEVİNİM: AFET SONRASI KONTEYNER YERLEŞİMLERİ

YÜKSEK LİSANS TEZİ

Hatice Seray ÖZBAY
(523221006)

Bilişim Anabilim Dalı

Mimari Tasarımda Bilişim Programı

Tez Danışmanı: Doç. Dr. Sema ALAÇAM

EYLÜL 2025

Hatice Seray ÖZBAY, a M.Sc. student of ITU Graduate School student ID 523221006, successfully defended the thesis/dissertation entitled "A MULTI LAYER CELLULAR AUTOMATA TO PARTICIPATORY ARCHITECTURE: POST-DISASTER CONTAINER SETTLEMENTS", which she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Assoc. Prof. Dr. Sema ALAÇAM** İstanbul Technical University	....
Jury Members :	**Assoc. Prof. Dr. Ethem GÜRER** İstanbul Technical University	....
	Prof. Dr. Özgür Mehmet EDİZ Uludağ University	....

Date of Submission : 9 September 2025
Date of Defense : 23 September 2025

__Anneme ve babama,__

FOREWORD

Firstly, I owe my deepest gratitude to my thesis advisor, Assoc. Prof. Dr. Sema ALAÇAM, for her invaluable guidance, continuous support, and encouragement throughout my master's studies. Her expertise, patience, and insightful feedback have greatly contributed to the development and completion of this thesis. I feel truly privileged to have had the opportunity to work with her.

I am profoundly grateful to the jury members, Assoc. Prof. Dr. Ethem GÜRER and Prof. Dr. Özgür Mehmet EDİZ. Their insightful contributions and constructive feedback have played a key role in shaping and enriching this thesis, while guiding me to consider my work from multiple perspectives.

With all my heart, I thank my beloved mother, Semra ÖZBAY, and my beloved father, Recep ÖZBAY, who have always nurtured my desire to learn and encouraged me from my childhood to pursue science, research, and learning. Without their love, patience, and limitless support, this journey would have been incomplete.

I am endlessly grateful to my fiancé, Berkan ÜNAL. He has given me courage and strength at every moment of my life. His patience and support have made this journey far more meaningful and valuable.

In addition, I wish to convey my deepest gratitude to my supervisors and colleagues at my workplace who have always supported my academic career and eased the challenge of studying while working.

Finally, I would like to offer my deepest thanks and affection to all my friends who have walked this path with me, shared in my journey, and never withheld their support.

September 2025 Hatice Seray ÖZBAY
(Architect)

TABLE OF CONTENTS

Page

ABBREVIATIONS

AR	: Augmented Reality
BFS	: Breadth-first Search
CA	: Cellular Automata
GUI	: Graphical User Interface
ML	: Multi Layer
MLCA	: Multi Layer Cellular Automata
MA	: Max Attempts
MUA	: Max Upper Attempts
NGO	: Non-governmental Organizations
2D	: Two-dimensional
3D	: Three-dimensional

LIST OF TABLES

Page

LIST OF FIGURES

Page

A MULTI LAYER CELLULAR AUTOMATA TO PARTICIPATORY ARCHITECTURE: POST-DISASTER CONTAINER SETTLEMENTS

SUMMARY

Changing sociological and economic conditions worldwide have pushed users into a position where they also take part in the design phase of the products they will use. Today, the user, in addition to being the end-user of a product, also wishes to contribute to its design and decision-making process, thus no longer remaining merely a beneficiary but also becoming a participant in the design. In this context, the collaborative design approach includes the user in the design process and incorporates them into the decision-making mechanism. This situation is also valid for the architectural design process. The architectural design product's user no longer wants to be merely the owner or user of the product but to take part in the design and decision-making process of the space they will live in or spend time in. The approach that involves the user in the architectural design process is called participatory architecture. Participatory architecture emerged in Europe in the 1950s and aims to produce more sustainable and effective projects in a social, environmental, and economic context by including the user in the design. With many examples around the world, the participatory architecture movement advocates a more democratic approach to the architectural design process by considering the needs and choices of the user. Participants involved in the process may belong to various groups such as project owners, NGOs, academics, students, professionals in the field, or local people. These stakeholders can contribute to the process in many ways. In traditional examples of participatory architectural design, users may participate to the design through methods such as interviews, surveys, and seminars. With these methods, the participant can be involved in every phase of the process, including the pre-design process. In some examples, the user even participates in the post-design construction phase. Participation can be enabled through various tools. Depending on the designer's preference, architectural drawings, sketches, or 3D models can be used. With the development of digital design tools, the concept of participation has become more interactive. Users without technical knowledge in the field can now see the outcomes of decisions simultaneously through various digital design tools. Interactive 3D models and simulations developed in the form of interactive games are examples of these. Within the scope of this thesis, it is proposed to build a bridge between generative design systems and participatory architecture and to integrate the user decisions as a data to the beginning of the workflow so that the output can be shaped by the user directly. During this process, a GUI was used to collect data from the user. Through the GUI, the user makes their selections and exports them to the designer, who then shapes the subsequent stages of the design process with their professional expertise.

As a design problem, the situation selected was the idle condition of container cities—originally established to provide temporary housing for earthquake victims—after their residents moved to permanent homes following the 2023 earthquakes centered in Kahramanmaraş/Türkiye. These container cities remain idle or are recycled after the temporary usage period ends. Due to the destructive earthquakes in the region, there is a need for spaces that will support the area socially. For this reason, it is aimed to

create alternative settlement areas by repurposing vacated container cities with the inclusion of the local population through participatory design principles.

During the development of this process, cellular automata algorithms, which are among generative design systems, were used. Cellular automata is a generative design system based on cells in a grid behaving according to rules defined by the states of their neighbors.

Within the scope of the thesis, the research question selected is whether a solution can be provided to create a new living space with the use of idle containers that emerged after the earthquake, through a process in which the local community actively participates, using settlement options generated by cellular automata.

In the implementation phase, the needed spaces and functions in the area were first evaluated. Since the primary purpose of the container cities has changed from providing shelter to serving as living spaces with social areas—due to the relocation of former residents to permanent homes—the following areas were determined as potential functions: health facilities such as hospitals and clinics, social spaces including dining areas, libraries, laundries, markets, and guest houses to meet temporary accommodation needs. In addition to these, open spaces for markets and parks were also planned. It was assessed whether the mentioned spaces required semi-open or open areas. Moreover, decisions were made regarding which functions should be placed on which floors. While ensuring that the most publicly accessible areas were positioned on the ground floor, it was ensured that the most private function—the guest houses—were placed on the top floor. This process of discovery subsequently laid the foundation for the introduction of the 'Functional' and 'Spatial' layers into the project workflow.

To create and run the cellular automata algorithm, the Rhinoceros Grasshopper program, which also provides 3D results, was used. Rules were defined in accordance with cellular automata principles, and the placement of containers was made to comply with these rules. The Python programming language was used, and the code was integrated into the model using the Grasshopper Python component. Thus, the Formal Relations layer, which is one of the layers of the MLCA, was established. The MLCA aimed to create settlement units that vary based on the possession of semi-open and open spaces appropriate to their functions. One of the original aspects of the thesis is the creation of a Functional Layer by searching for functional meaning in the formal outputs produced by the cellular automata algorithm. Afterwards, specific functions were assigned to the forms identified in the mass studies. Also, the relationships between these functions were examined, leading to the creation of the Spatial Layer. Thus, the developed generative design system, as a multi layer cellular automata mechanism, aims to offer alternative living spaces for container cities that have become idle after their residents moved into permanent housing. Since the configurations generated by the MLCA are analyzed based on the question of whether they form meaningful spaces such as semi-open areas, courtyards, and transition zones where users can live—not just random arrangements—this aspect offers a human-centered and context-sensitive design proposal beyond a purely algorithmic design. Since the alternatives produced by the MLCA system are created according to the choices of the local community, the method offers a generative and participatory alternative to the design problem.

KATILIMCI MİMARLIKTA ÇOK KATMANLI HÜCRESEL ÖZDEVİNİM: AFET SONRASI KONTEYNER YERLEŞİMLERİ

ÖZET

Dünya genelinde değişen sosyolojik ve ekonomik koşullar, kullanıcıları yalnızca ürünlerin nihai tüketicisi olmaktan çıkararak, aynı zamanda bu ürünlerin tasarım sürecine de dahil olmaya yöneltmiştir. Günümüzde kullanıcı, bir ürünün sadece son kullanıcısı olmakla yetinmemekte; tasarım ve karar alma süreçlerine de katkı sunmak istemektedir. Böylece yalnızca faydalanıcı değil, aynı zamanda tasarıma katılan bir paydaş haline gelmektedir. Bu bağlamda, işbirlikçi tasarım yaklaşımı kullanıcıyı tasarım sürecine dahil ederek karar alma mekanizmasına entegre etmektedir. Bu durum, mimari tasarım süreci için de geçerlidir. Mimari tasarım ürününün kullanıcısı artık yalnızca yapının sahibi veya kullanıcısı olmakla yetinmemekte; yaşadığı veya vakit geçireceği mekânın tasarım ve karar süreçlerinde yer almayı tercih etmektedir. Kullanıcının mimari tasarım sürecine dâhil edilmesi yaklaşımı 'katılımcı mimarlık' olarak adlandırılmaktadır. Katılımcı mimarlık, 1950'li yıllarda Avrupa'da ortaya çıkmış olup kullanıcıyı tasarıma dahil ederek sosyal, çevresel ve ekonomik bağlamda daha sürdürülebilir ve etkin projeler üretmeyi amaçlamaktadır. Dünyanın birçok yerinde örnekleri bulunan katılımcı mimarlık hareketi, kullanıcı ihtiyaçlarını ve tercihlerini göz önünde bulundurarak mimari tasarım sürecinde daha demokratik bir yaklaşımı savunmaktadır.

Sürece dahil olan katılımcılar; proje sahipleri, sivil toplum kuruluşları, akademisyenler, öğrenciler, alanında uzman profesyoneller ya da yerel halktan oluşabilmektedir. Bu paydaşlar sürece farklı şekillerde katkı sunabilmektedir. Geleneksel katılımcı mimarlık örneklerinde, kullanıcılar mülakatlar, anketler ve seminerler gibi yöntemlerle tasarıma katılabilmektedir. Bu yöntemlerle katılımcılar ön tasarım aşaması da dahil olmak üzere sürecin her evresinde yer alabilirler. Bazı örneklerde kullanıcılar, tasarım sonrası inşaat aşamasına dahi katılım göstermektedir. Katılım farklı araçlarla mümkün kılınabilmektedir. Tasarımcının tercihine bağlı olarak mimari çizimler, eskizler veya üç boyutlu modeller kullanılabilmektedir. Dijital tasarım araçlarının gelişmesiyle birlikte katılım kavramı daha da etkileşimli bir hâl almıştır. Artık teknik bilgisi olmayan kullanıcılar bile çeşitli dijital tasarım araçları sayesinde alınan kararların sonuçlarını eşzamanlı görebilmektedir. Etkileşimli üç boyutlu modeller ve oyun formunda geliştirilen simülasyonlar buna örnek olarak verilebilir.

Bu tez kapsamında, üretken tasarım sistemleri ile katılımcı mimarlık arasında bir köprü kurulması ve kullanıcı kararlarının veriye dönüştürülerek iş akışının başlangıcına entegre edilmesi önerilmektedir. Böylece çıktı doğrudan kullanıcı tarafından şekillendirilebilecektir.

Bu süreçte kullanıcıdan veri toplamak için bir grafiksel kullanıcı arayüzü (GUI) kullanılmıştır. GUI üzerinden tercihlerde bulunan kullanıcı, bu seçimleri dışa aktararak (export) tasarımcıya iletir; tasarımcı ise kendi profesyonel bilgisiyle tasarım sürecinin sonraki aşamalarını şekillendirir. Arayüz tasarlanırken özellikle dikkate alınan nokta, katılımcının konuda profesyonel ve teknik bilgiye sahip olmadığıdır. Bu sebeple, kelime ve görseller oldukça sade ve herkes için anlaşılır biçimde hazırlanmıştır. Arayüzün, kullanıcı kararlarını veriye dönüştüren ve bu verileri tasarımcıya aktaran bir köprü olarak işlevi görmesi amaçlanmıştır. Bu sayede teknik bilgisi olmayan kullanıcılar dahi sürecin başlangıcına dahil olmuş, ortaya çıkan tasarım alternatifleri kullanıcı tercihleri doğrultusunda şekillenmiştir.

Tasarım problemi olarak, 2023 yılında merkez üssü Kahramanmaraş olan depremler sonrasında depremzedeler için geçici barınma amacıyla kurulmuş olan konteyner kentlerin, kalıcı konutlara taşınılmasının ardından atıl duruma geçmesi seçilmiştir. Bu konteyner kentler, kullanım süresi dolduktan sonra ya atıl bırakılmakta ya da geri dönüştürülmektedir. Bölgede yaşanan yıkıcı depremler nedeniyle sosyal yaşamı destekleyecek mekânlara ihtiyaç duyulmaktadır. Bu nedenle, boşalan konteyner kentlerin katılımcı tasarım ilkeleri doğrultusunda yerel halkın dahil edilmesiyle yeniden işlevlendirilerek alternatif yerleşim alanlarına dönüştürülmesi hedeflenmektedir.

Bu sürecin geliştirilmesinde üretken tasarım sistemlerinden biri olan hücresel özdevinim (cellular automata) algoritmalarından yararlanılmıştır. Hücresel özdevinim, belirli kurallara göre komşu hücrelerin durumlarını esas alarak çalışan ızgara tabanlı bir üretken tasarım sistemidir. Basit hücresel etkileşimler sonucunda karmaşık mekânsal düzenler ve farklı yerleşim senaryoları üretilebilmektedir. Bu algoritmayı kullanarak hayata geçirilen mimari tasarım örnekleri mevcuttur.

Tez kapsamında araştırma sorusu; deprem sonrasında atıl duruma geçen konteynerlerin, yerel halkın aktif katılımıyla ve hücresel özdevinim algoritmasının ürettiği yerleşim alternatifleri üzerinden değerlendirilerek, yeni bir yaşam alanı oluşturulup oluşturulamayacağıdır.

Uygulama aşamasında öncelikle bölgedeki ihtiyaç duyulan mekânlar ve işlevler belirlenmiştir. Konteyner kentlerin temel işlevi, sakinlerinin kalıcı konutlara taşınılmasıyla birlikte barınma olmaktan çıkıp, sosyal alanları olan yaşam mekânlarına dönüşmüş olduğundan; sağlık yapıları (hastane, klinik), sosyal alanlar (yemekhane, kütüphane, çamaşırhane, market), geçici konaklama ihtiyacını karşılayacak misafirhaneler ve açık alanlar (pazar yerleri, parklar) teklif edilen konteynerlar yerleşimlerinin potansiyel işlevleri olarak tanımlanmıştır. Bu mekânların açık, yarı açık ya da kapalı olması gereklilikleri değerlendirilmiştir. Ayrıca hangi işlevlerin hangi katlarda yer alacağına karar verilmiştir. En kamusal alanlar zemin kata, en özel işlev olan misafirhaneler ise en üst kata yerleştirilmiştir. Bu keşif süreci, projeye "Functional" ve "Spatial" katmanların eklenmesine temel oluşturmuştur.

Hücresel özdevinim algoritmasının oluşturulması ve çalıştırılmasında üç boyutlu sonuçlar da üreten Rhinoceros Grasshopper programı kullanılmıştır. Hücresel özdevinim prensiplerine uygun kurallar tanımlanmış, konteyner yerleşimleri bu kurallara uygun biçimde gerçekleştirilmiştir. Python programlama dili kullanılarak kod yazılmış ve Grasshopper'ın Python bileşeni aracılığıyla modele entegre edilmiştir. Böylece MLCA'nın katmanlarından biri olan "Formal Relations" katmanı oluşturulmuştur. Tezin özgün yönlerinden biri, hücresel özdevinim algoritmasının ürettiği biçimsel çıktılarda işlevsel anlamların tasarımcı tarafından aranması ve böylelikle "Functional Layer"ın üretilmesidir. Sonrasında, kütle çalışmalarıyla tanımlanan biçimlere belirli işlevler atanmış ve bu işlevler arasındaki ilişkiler incelenerek "Spatial Layer" oluşturulmuştur. Böylece geliştirilen üretken tasarım sistemi, çok katmanlı hücresel özdevinim mekanizması aracılığıyla, kalıcı konutlara taşınılmasının ardından atıl duruma geçmiş konteyner kentler için alternatif yaşam alanları önermektedir.

MLCA tarafından üretilen kurgular yalnızca rastlantısal düzenlemeler değil; kullanıcıların yaşayabileceği yarı açık alanlar, avlular ve geçiş mekânları gibi anlamlı mekânlar üretip üretmediği üzerinden değerlendirilmektedir. Bu yönüyle, yalnızca algoritmik bir tasarım olmaktan öte, insan odaklı ve bağlama duyarlı bir tasarım önerisi sunmaktadır. Ayrıca, MLCA sistemi tarafından üretilen alternatifler yerel halkın tercihleri doğrultusunda şekillendiğinden, yöntem söz konusu tasarım problemine üretken ve katılımcı bir alternatif sunmaktadır.

1. INTRODUCTION

Following large-scale natural disasters such as earthquakes, container-based settlements have become a widely used strategy to provide rapid shelter for displaced populations. While these modular settlements offer advantages in terms of fast deployment, they often lack spatial cohesion, adaptability, and long-term social integration. Once the emergency phase ends, many containers become idle or are underutilized. A large number of containers rapidly transported to the area during the crisis turn into dysfunctional elements in the urban fabric. Similar situations were observed after the Mw 7.7 and Mw 7.6 magnitude 6 Şubat 2023 earthquakes (AFAD, 2023) centered in Kahramanmaraş/Türkiye. The containers deployed to the region remained unused once residents moved into permanent housing or left the area.

This research arises from the necessity to reconceptualize these containers—not as static solutions, but as dynamic architectural components that can evolve through computational design and participatory architecture, involving both the local community and other stakeholders in the process.

The aim of this study is to generate container settlement models that are both computationally adaptive and socially participatory, using MLCA. By integrating the functional requirements of containers as distinct layers within the CA system, the goal is to produce spatial arrangements that are not only efficient and physically feasible but also responsive to human needs and meaningful in design.

The research methodology is based on a 3D CA model developed in the Rhino-Grasshopper environment using the Python programming language. This model simulates the placement of container units on a modular grid, considering horizontal connections, vertical stacking, and functional constraints. Spatial tendencies (such as preferred proximities, shared space usage, and form-function relationships) are embedded into the system as rules. In this way, the generated settlement scenarios respond to both formal and non-formal dynamics, shaping the spatial forms through not only rule-based computation but also an embedded social layer.

As indicated in Figure 1.1, the flow of the research can be divided into two main phases. The first part is the generation of container mass layouts automatically produced by the MLCA through coding. In this phase, the GUI which was developed to collect data from the user in order to ensure participation is used at the initial stage, and the “Formal Relations” layer, which is one of the MLCA’s layers, is defined. Subsequently, in the manual phase, the other two layers of the MLCA, namely “Functional Relations” and “Spatial Relations,” are integrated into the project through the intervention of the designer. If the generated model possesses the desired features, it is then further detailed with structural and other secondary elements to be prepared for practical application.

The core hypothesis of the research is that a multi layer cellular automata system, when integrated with functional requirements and semantic inputs, can produce container settlements that are structurally coherent, socially adaptable, and spatially meaningful. The outcome aims to incorporate the principles of participatory architecture by involving local residents in the design process. This method reconceptualizes container settlements not solely as modular units for temporary dwelling, but as instruments for collaborative production, thus providing a design framework consistent with the principles of participatory architecture.

The scope of this thesis focuses on proposing new settlement models for the containers left idle after the Kahramanmaraş earthquakes, particularly emphasizing their reuse.

Accordingly, the research question is defined as follows:

"Can MLCA, integrated with functional requirement inputs, generate new living environments and meaningful spatial arrangements with post-earthquake idle containers, operating under the principles of participatory architecture?"

By addressing this question, the research aims to present an example that combines computational design with participatory architecture, while also proposing that emergency shelter solutions can be repurposed in a sustainable and collaborative manner to acquire new architectural meaning beyond their initial temporary use.

Figures and tables were prepared by the author unless otherwise stated.

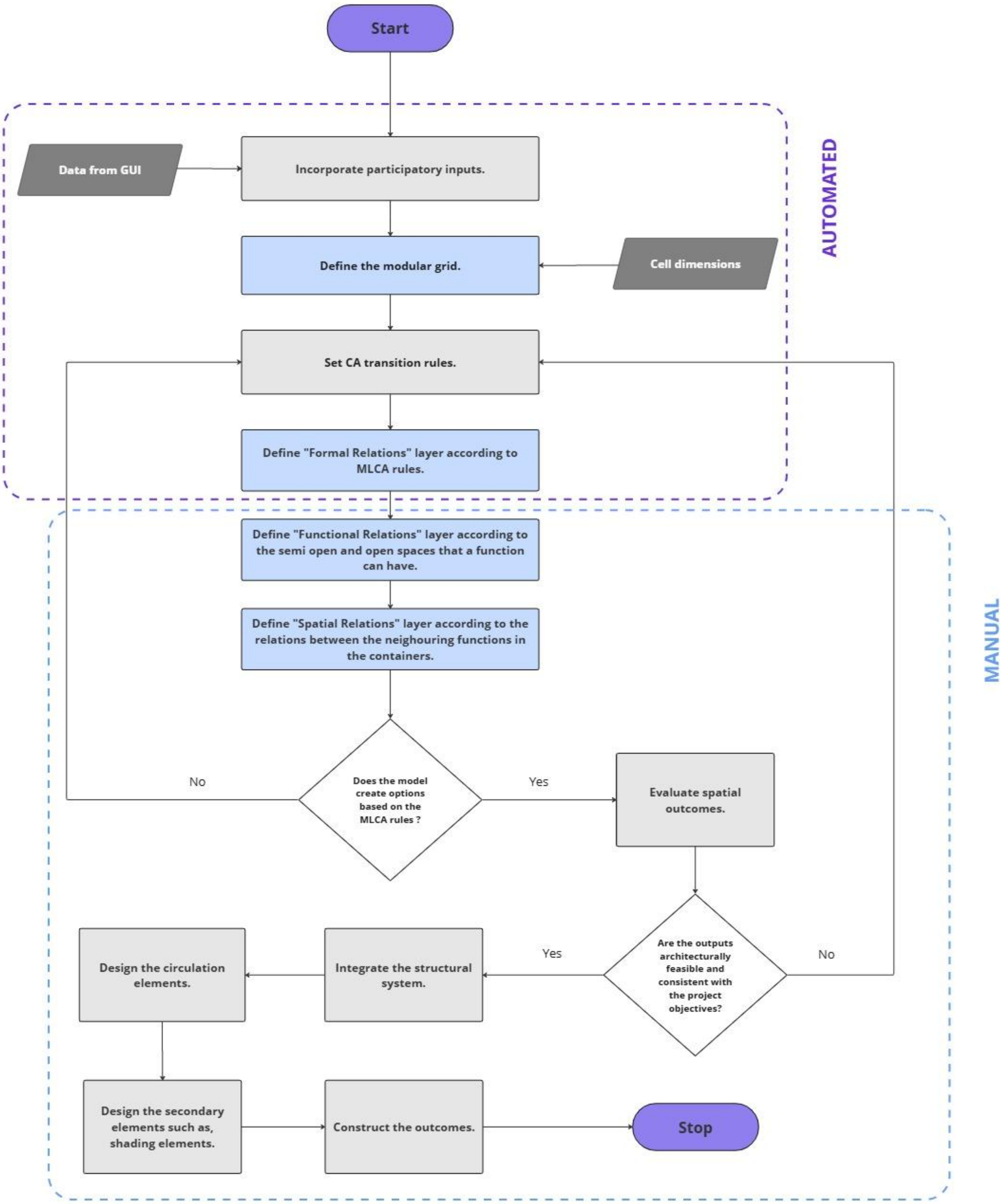


Figure 1.1. Flowchart.

2. PARTICIPATORY ARCHITECTURAL DESIGN TOOLS AND METHODS

The user is transforming into a participant who seeks to have a say in the process rather than just experiencing the final product in the architectural design process. Collaborative design processes allow users to be involved in design. French sociologist Henri Lefebvre divided the world into abstract and concrete realms. The abstract realm is where the design process is conducted by the designer, and the user is passive. The concrete realm is the real world where the user experiences the product. The intersection of these two separate realms forms a new area that includes the collaborative design process as displayed in Figure 2.1. In this area, the designer and the user are in communication and jointly create a product (Lefebvre, 1970/2003).

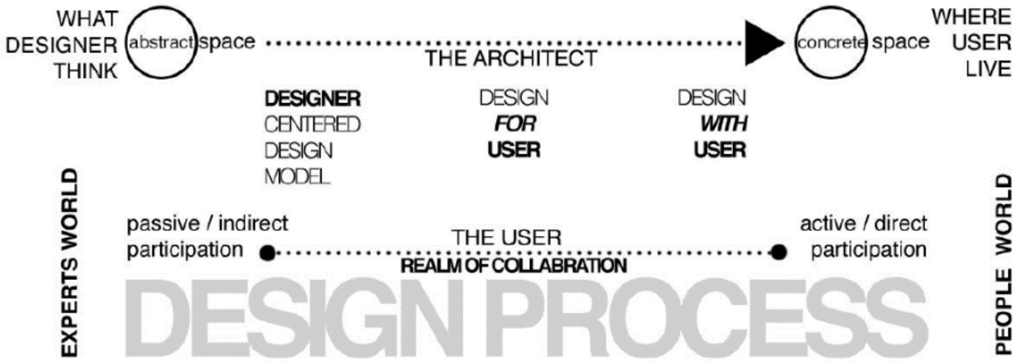


Figure 2.1: Collaborative design process diagram (Terlemez, 2018).

2.1 The Historical Development of Participatory Architecture

Shelter is recognized as one of the fundamental human needs. Even early humans, after securing food to satisfy their hunger, experienced the need to create enclosures as protective environments against external factors. Regarding the absence of tools, constructing such enclosures necessitated collective effort. The first humans addressing their shelter needs simultaneously acted as both designers and users, collaboratively and interatively shaping and constructing the environments in which they would live.

Lorenzon (2023), examines the development of earthen architecture in the Neolithic period within the framework of the concept of ‘communities of practice.’ It is emphasized that earthen architecture was shaped as a community-based practice, reflecting a participatory design and collaborative production approach. This is due to the labor required both in shaping the components and in their assembly. The advantages brought by this labor, together with communal living conditions, demonstrate that early humans adopted a collaborative and community-oriented approach in architectural processes. Therefore, since humans transitioned to a settled lifestyle, they have been both users and practitioners of collaborative design methods. As history progressed, with the emergence of human rights awareness and the desire for ownership, the concept of participatory architecture also began to take shape.

The participatory architectural approach that began in Europe in the 1950s aims to create more sustainable projects socially, environmentally, and economically by involving users in the design processes. Participatory architecture represents an approach that aims to unite the user and the designer in order to produce a common design product. This approach actively includes the needs and preferences of users, local communities, civil society organizations, and other relevant stakeholders in the architectural design process. Participatory architecture ensures that architectural projects are more sustainable in social, cultural, environmental, and economic contexts by making design decisions together with the community. At the end of the design process, the involvement of the users results in a project that belongs not only to the designer but to all stakeholders involved through a collaborative approach. The inclusion of participants in this process makes it democratic, leading to more accurate products from the user's perspective. Additionally, this process initiates an interactive educational experience for the participants (Terlemez, 2018). The change in the architectural understanding after the entrance of participatory architecture to the design practices includes a series of tools, methods, and new ideology in design research and for architectural practice (Luck, 2018).

When examining changes in digital design tools within the participatory architecture process, it becomes evident that with the development of CAD tools after 1980, these tools also began to be used in participatory architecture, enabling forward-looking projects. As moving towards the 2000s, the development of digital design tools accelerates with the use of generative design systems. In the early 2000s, Jose Duarte's project on Alvaro Siza's Malagueira Houses (Figure 2.2) can be considered an exemplary project. Duarte (2001), used shape grammars to mass-customize Alvaro Siza’s Malagueira housing project. The use of shape grammars in this project revealed the mathematical and grammatical structure of architectural design, allowing designers to create new and creative solutions. Additionally, Duarte’s approach demonstrated that using shape grammars according to user needs and lifestyles provides countless possibilities for design.

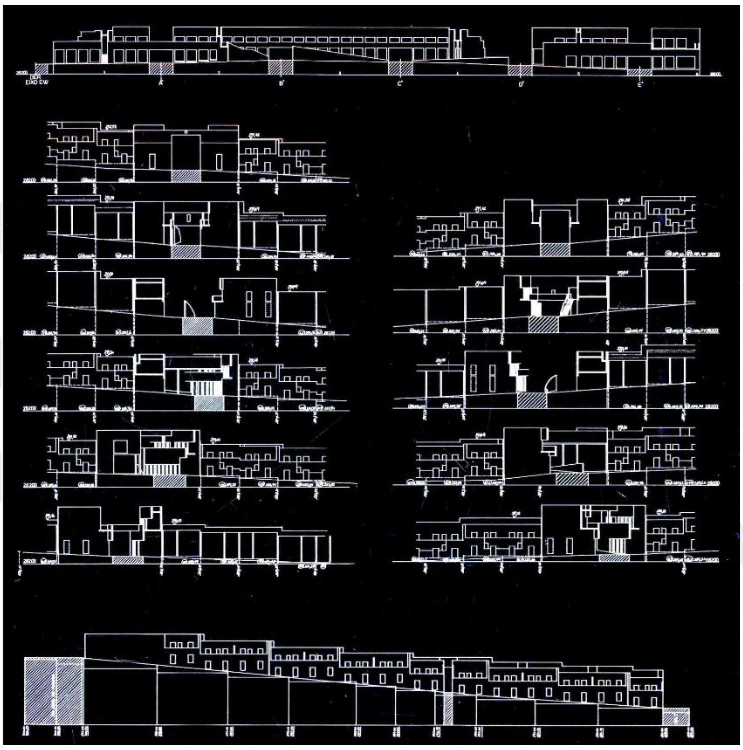


Figure 2.2: Sections of Alvaro Siza's Malagueira Houses (URL-5).

As moving on the timeline passing the 2010s, it is observed that projects developed at the MIT Lab aimed to ensure participation by incorporating the qualitative characteristics of the user. The Home Genome project, which used artificial intelligence algorithms, analyzes qualitative characteristics such as the user's lifestyle preferences, tastes, and the demographic structure of their household to offer suitable design examples to the user. While aiming to provide userspecific designs, it also intended to raise awareness by prompting users to reconsider their needs and preferences through the questions asked (Vardouli, 2012).

Today, participatory processes can be supported through generative design systems, artificial intelligence models, AR, VR, and numerous other digital tools. According to

Koçer Özgün and Alaçam (2023), findings from user testing suggest that integrating AR into the design process can enhance interaction between physical and digital environments. In particular, enriching physical models with AR applications transforms them into tools that not only support mobile design practices but also serve as alternative representation methods beyond conventional participation techniques.

Participatory design creates an environment where users can openly express their needs and preferences without remaining anonymous. At the same time, designers and experts become actively involved in the process by engaging with participants' ideas, which fosters a sense of trust and collaboration. This approach enables the collective creation of living spaces, supported by mutual cooperation and shared responsibility (Çelik, 2022).

In the following two sections, two projects, Responsive City Design Game and Augmented Reality Sandbox, that establish various means of interaction with participants and construct interactive design environments are examined.

Today, designers make use of technological developments and tools to create various forms of interaction with participants. Some of the main objectives here are to convey the subject to participants who do not possess technical knowledge in a comprehensible way and to create an interactive design environment that also appeals to the senses.

2.1.1 Augmented Reality Sandbox

A representative example of a project in which users without professional expertise in the field can learn through interaction and observe the outcomes of their decisions is the Augmented Reality Sandbox. The project was developed by the UC Davis W.M. Keck Center for Active Visualization in the Earth Sciences.

The project aims to engage museum visitors or members of the public in creating topographic forms using an augmented reality (AR) sandbox, thereby enhancing interaction and learning by overlaying visual simulations of elements such as contour lines and water flow. This technology merges users' physical interventions with digital simulations, making the learning process more interactive and intuitive. By enabling users to learn through direct manipulation, it offers an alternative to traditional educational methods. The AR Sandbox system operates through the physical shaping of sand, as seen in Figure 2.3, users form terrain features such as mountains, valleys, and riverbeds, which are scanned by a depth camera to generate a Digital Elevation Model. The corresponding topography is then projected back onto the sand with colored contours and water levels. By holding their hands above the surface, users can trigger simulated rainfall, which demonstrates how water flows across the terrain. This allows them to explore spatial processes through immediate visual feedback (Reed et al, 2016). Since users can instantly observe the outcomes of their design decisions, they become part of an interactive decision–response cycle. Consequently, the system serves as a translational interface, enabling individuals without technical expertise to engage meaningfully with geospatial concepts.

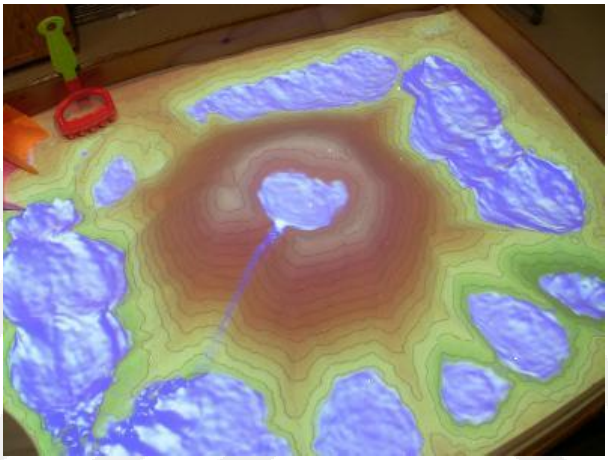


Figure 2.3: The AR Sandbox (URL-3).

2.1.2 Responsive City Design Game

The project titled "The Responsive City Design Game" by Ekim Tan and Juval Portugali (Tan and Portugali, 2012) presents a detailed account of an experimental urban design game structured within the framework of complexity theory. The game is set in a designated expansion area of Almere Haven, where fifteen participants (Figure 2.4) , each representing a distinct socio-demographic user profile, are tasked with placing 400 housing units. At the beginning of the process, only two basic organizational rules are introduced to participants, with no predetermined planning principles or design norms provided. The design progresses in a turn-based sequence, where each participant makes spatial decisions by considering both the needs of the represented user profile (e.g., elderly couple, single individual, family with children) and the existing spatial context shaped by previous interventions.

During the design process, spatial voids and building clusters that emerge are identified by participants as public spaces (e.g., squares, parks), allowing the configuration of urban space to develop organically through the game's dynamics. In the later stages of the process, several secondary rules, not defined at the outset, are proposed by participants and adopted collectively through group discussion. These rules are then used as reference points for subsequent placement decisions and play a role in shaping the evolving spatial layout. Conflicts arising from the differing spatial demands and interests of the various user profiles are addressed through negotiation, discussion, and compromise among participants.

Figure 2.4: Players during the simulation (Tan and Portugali, 2012).

The resulting urban pattern displays a non-uniform but cohesive spatial configuration, including self-emerging centralities, housing clusters, and public spaces. The authors highlight that design decisions in this experiment do not follow a linear or top-down planning model; rather, they evolve through interaction among participants, feedback loops, and mutual adaptation. The design game illustrates how urban space, viewed as a complex system, can emerge not from fixed regulations but from user-driven decision-making processes. In this context, the study explores the applicability of complexity theory to urban planning and demonstrates the potential of user representation, negotiation-based decision-making, and collective spatial organization in the production of the built environment.

As demonstrated in this example, user participation and interaction are not limited to the architectural scale but can also be applied at the urban scale. By providing users with defined boundaries, the process can become more interactive. In this way, the participatory design process evolves not as a linear sequence, but rather as a cycle shaped by feedback and negotiation.

2.2 Participatory Architecture in Post-Disaster Contexts

Throughout the history, some of the most effective examples of participatory architecture are housing projects developed with earthquake survivors who lost their homes due to natural disasters. The goal in post-disaster projects is to introduce a design and prefabrication process for mass-customized emergency housing. By simplifying construction details, disaster victims can actively participate in the reconstruction process. This approach accelerates building speed and reduces labor costs, while also considering the psychological well-being of the victims (Rezoug, 2013). These projects aim not only to provide rapid housing solutions tailored to the preferences of the affected individuals but also to manage the transition from temporary prefabricated shelters to permanent dwellings through the principles of participatory design. During this process, users may be more inclined to embrace the principle of collaboration, as they experience feelings of solidarity, belonging, and unity.

While certain studies highlight the importance of involving local communities in shelter construction and utilizing local materials, the literature provides little insight into systematic ways of implementing these ideas. This gap creates an opportunity to explore how sustainable practices and participatory and collaborative strategies might enhance both the cultural appropriateness and overall effectiveness of temporary housing. Expanding on these approaches is essential to ensure that post-disaster shelters not only address immediate needs but also contribute to strengthening community resilience in the long term (Makadi et al, 2025).

The Villa Verde project and the Düzce Umut Evleri, used as examples in the following section, were designed as instances of post-disaster participatory architecture.

2.2.1 Düzce Umut Evleri

An exemplary project that applies the principles of participatory architecture in a post-disaster context is the Düzce Umut Evleri (Düzce Hope Homes) project (URL-1). The Project was launched in 2015 by the participants and volunteers of the Düzce Umut Atölyesi. It emerged as a grassroots response to the long-standing housing challenges faced by survivors of the 1999 Marmara Earthquake, many of whom remained in inadequate and temporary shelters for over a decade. The primary objective of the project was to develop a new mass housing model in Düzce that prioritized the active involvement of future residents, ensuring that their needs, preferences, and lived experiences shaped the outcome of the built environment.

The initiative brought together a multidisciplinary and participatory working group composed of architects, academics, university students, planners, and, most importantly, the future homeowners themselves. This collective operated a collaborative approach throughout the design process. Over the course of a year, the group engaged in a comprehensive and inclusive design process, organizing a series of workshops, site visits, interviews, and meetings with the residents. These sessions not only served to gather input but also functioned as platforms for mutual learning and dialogue between experts and non-experts.

To bridge the gap between architectural knowledge and user experience, the design team employed a variety of participatory tools and techniques. These included collaborative games such as plot division and space placement exercises, which allowed participants without prior design experience to meaningfully contribute to decisions regarding site layout, spatial organization, and housing typologies. Representation tools such as hand-drawn sketches, physical models, and digital simulations were used to visualize different scenarios and facilitate discussion among the stakeholders.

Many collective interviews conducted with homeowners highlights the project's commitment to ensuring that users were not passive recipients of predesigned solutions, but rather co-creators of their future living environments. This model of engagement underscored the transformative potential of participatory architecture in fostering a sense of ownership, empowerment, and community resilience.

2.2.2 Villa Verde project

On February 27, 2010, Chile was hit by an 8.8-magnitude earthquake followed by a devastating tsunami, resulting in widespread destruction throughout the country. Villa Verde is a significant social housing project built in Chile in 2010, designed to address the housing needs of middle- and lower-income segments of society and create a housing project based on the idea of incremental housing (URL-2). The project was conceived as a response to the ongoing demand for affordable housing in the country, with the intention of providing a flexible model that could accommodate the varying economic capacities of its future residents. The intended recipients comprised both corporate employees and individuals displaced by a recent earthquake. In this scenario, the 'half a good house' strategy was transitioned from slum upgrading initiatives to a public-private housing project, integrating a post-disaster element (Aravena and Iacobelli, 2016). The housing units were divided into two parts: one half was fully designed and constructed by architects, while the remaining portion was left intentionally unfinished. The initial area in each unit is 57 square meters, reaching up to 85 square meters after completing the extensions. The unfinished portion allowed homeowners to complete the construction of their homes over time, depending on their financial resources and preferences (Figure 2.5). The decision to leave these units unfinished reflected the project's adaptability to the users' economic conditions, allowing for a more personalized and gradual approach to housing development.

Figure 2.5: User interventions (O'Brien et al, 2020, p. 350).

The Villa Verde project involved two primary stakeholders: the architects, who were responsible for the design of the finished units, and the homeowners, who would eventually occupy the homes. Notably, the design process did not involve direct communication between the architects and the future residents during the initial stages. However, toward the later stages of the project, homeowners were given the opportunity to influence the design and construction process, specifically in terms of finishing the unfinished portions of their homes. While this involvement did not occur through formalized or direct interaction with the architects during the conceptual design phase, it allowed for a degree of user agency in shaping the final outcome of the housing units. This model of participation highlights an important distinction in participatory design approaches, where users are involved in later stages of construction rather than at the inception or design phases.

In the case of Villa Verde, no formal representation tools such as sketches, models, or digital simulations were employed to facilitate communication between users and designers. The absence of direct communication between the two groups meant that the architects did not rely on these traditional participatory tools. Instead, users were integrated into the construction process in a more pragmatic manner, with the final stages of the project leaving space for homeowners to personalize their living spaces based on their available resources. This model challenges traditional participatory methods that emphasize constant dialogue and engagement throughout the design process, yet it still acknowledges the importance of user involvement, particularly in post-construction phases. By empowering homeowners to gradually complete their homes, the Villa Verde project underscores the potential of participatory housing initiatives to accommodate users' evolving needs and capacities over time.

As illustrated in both of the aforementioned cases, users can be integrated into the architectural process at multiple stages, including pre-design, design, post-design, and even construction. It is crucial to acknowledge that these participants, often the end users of the built environment, generally do not possess specialized knowledge or professional training in architecture, design, or construction. Their involvement, therefore, necessitates a mediating mechanism that can translate professional language, technical data, and spatial representations into more accessible and comprehensible forms. At this point, digital design tools play a significant role as mediators in the communication between professional designers and users. They act as translational mediums that enable knowledge exchange and collaborative decision-making between parties with differing levels of expertise. These tools, when designed appropriately, can democratize the design process by allowing non-experts to meaningfully engage with spatial ideas and contribute to the design. The outputs of such digital tools may include 3D models, two-dimensional drawings, visualizations, and interactive simulations, all of which serve as communicative artifacts that bridge the cognitive and experiential gap between designers and users. By doing so, they allow users to visualize, evaluate, and even modify design proposals in real time, often in participatory workshops or collaborative platforms.

In the subsequent sections, specific examples will be presented to demonstrate how digital design tools have been utilized to facilitate interaction with users who lack technical knowledge of the professional domain.

2.2.3 Pono village

In Pakistan, under the leadership of architect Yasmeen Lari, numerous projects exemplifying collaborative design have been carried out. The country frequently faces increasingly severe rains and floods. Lari argues that reinforced concrete structures are neither resilient to these climatic events and disasters nor appropriate for the region. In the most recent floods (Figure 2.6), the collapse of such buildings caused the deaths of thousands of people.

Figure 2.6: Pono village under flood (URL-7).

She also emphasizes that concrete, due to its high heat retention capacity, is unsuitable for Pakistan's hot summer climate. In addition, since the production of concrete and fired bricks releases significant amounts of carbon dioxide, it further contributes to the greenhouse effect that exacerbates flooding disasters. Moreover, studies of concrete houses in villages revealed that once construction was completed, homeowners struggled with the sustainability of the materials and faced difficulties expanding or reducing their houses as needed. According to Lari, the solution lies in existing traditional methods and in designing structures adapted to local conditions (URL-6).

Established by Lari in 1980, the Heritage Foundation of Pakistan provides training for villagers in Sindh province to construct their own flood-resilient dwellings using cheap, locally sourced, and low-carbon materials. Her architectural approach incorporates bamboo panels placed on elevated platforms stabilized with earth and lime, with design adjustments that enhance the water resistance of traditional mud huts. After gaining the necessary skills, residents are able to expand their communities and pass on the knowledge to others. Between mid-September and the end of 2022, the foundation supported the construction of 3,500 homes across 60 villages. Lari's vision extends beyond promoting alternative building materials; she seeks to redefine the entire post-disaster relief framework. In Pono village, Sindh, a group of ten certified craftsmen from the foundation trains workers to reinforce soil, fabricate bamboo panels, assemble them into octagonal shelters, and install roofing. These craftsmen subsequently conduct three-day workshops in other villages, guiding residents through the construction process. Once households can complete a basic one-room bamboo shelter, they are taught how to transform it into a permanent residence using earth and lime, personalize it with decorative façades, and adapt the technique for larger communal buildings such as schools or meeting halls. According to Lari, villages can then link with neighboring communities to disseminate these construction skills.

In Pono Eco Village, approximately 1,000 families have built a resilient community by employing low-carbon techniques that combine bamboo, clay, and lime (URL-7).

Figure 2.7: Training center in the Pono village (URL-8).

Beyond housing, the foundation built a training center (Figure 2.7). In the center, additional training sessions that teach villagers to produce clay tiles, cooking stoves, woven mats, and similar items—skills and products that can be exchanged with nearby communities are organized. Even prior to the catastrophic floods of 2022, Lari's team had assisted in the construction of 950 houses in Sindh, using prefabricated bamboo panels assembled on site and larger structures within Pono village. Post-flood inspections revealed that these buildings, which had remained submerged for two months, sustained no structural damage. According to Lari, each shelter costs less than $200 to build, covering material procurement and trainers' wages. Furthermore, she facilitates the establishment of village committees—often under the leadership of local women—to channel investment into micro-industries that strengthen community resilience and economic independence (URL-6).

The projects examined in terms of user participation and interaction are presented comparatively in Table 2.1 under the headings of subject, interaction/participation tools and participants.

Table 2.1: Collaborative and interactive practices.

Project	Subject	Participants	Interaction/ Participation Tools
Augmented Reality Sandbox	Geology	Museum visitors, Non-professionals	AR, 3D Simulation, Sandbox
Responsive City Design Game	Architectural and urban design	15 graduate students with diverse cultural and disciplinary backgrounds	Mock-ups, Architectural drawings
AR-based GUI	Architectural design (mixed used)	5 graduate students	GUI, AR environment
Malagueira Houses	Architectural design (housing)	House owners, archictect	Hybrid use of video conferencing, chat systems, e-mail, web-based sharing platforms, and physical/digital drawing environments
Düzce Umut Evleri	Architectural design (mixed used)	Home owners	Hand-drawn sketches, physical models, and digital simulations
Villa Verde project	Architectural design (mixed used)	Home owners	No direct communication between the user and the designer
Pono village	Architectural and urban design	Local communities, trained local craftsmen, NGO, donors and funding organizations, local government authorities	Construction workshops, physical models, learning by making

3. CELLULAR AUTOMATA AND ARCHITECTURAL IMPLEMENTATIONS

Computational frameworks allow designers to generate various design languages instead of manually creating individual designs. The actual design processes are performed and (semi-) automated by the system based on the design knowledge or constraints provided by the designers, enabling real-time interaction with the system. To facilitate design automation, multiple computational methods have been developed. These generative design systems range from simple rule-based models that produce random results to sophisticated AI-driven systems that aim to emulate certain aspects of human design thinking. Among the commonly employed generative design algorithms in the literature are shape grammars, L-systems, genetic algorithms, and cellular automata. While these algorithms share some similarities, they also differ in significant ways. Consequently, various generative design systems are better suited for addressing specific design challenges (Gu et al, 2010). A notable example of generative design systems is the bBox which was developed using the Unity 3D engine. This system allows for the rapid production of container-based settlements that can respond to different real-life scenarios such as ecological, agricultural, and post-disaster in line with predefined rules and parameters such as population, climate, budget, and geography. The model demonstrates how rule-based and procedural generation approaches can be integrated into interactive design frameworks, enabling real-time solutions to complex architectural problems (Şen-Bayram et al., 2023).

Within the scope of this thesis, focusing on CA from generative design systems, the produced model will be utilizing this design system.

CA is a computational and generative approach that models growth by representing complex systems through simple units governed by basic rules (Krawczyk, 2002). It was first discovered and developed in the late 1940s by John von Neumann and Stanislaw Ulam. In this work, von Neumann (1966) theoretically proposed the first comprehensive cellular automaton model and demonstrated how a cell could replicate itself. CA is capable of producing complex behaviors from a relatively simple rule set, making them well-suited for simulating intricate systems such as urban growth, wildfire spread, disease transmission, and traffic flow (Jiao and Boerboom, 2006).

3.1 Components of Cellular Automata

The foundational structure of cellular automata, as developed by Ulam, von Neumann, and expanded by Wolfram (1984), includes five main elements: the lattice, neighborhood, cell states, transition rules, and time.

- Lattice (Grid): In CA, cells are positioned within a defined grid system. This grid is used to determine the location of each cell and its neighborhood relationships. Rather than being arranged in a one-dimensional line, cells can also be positioned on 2D or 3D lattices as displayed in Figure 3.1 (Wolfram, 2002).

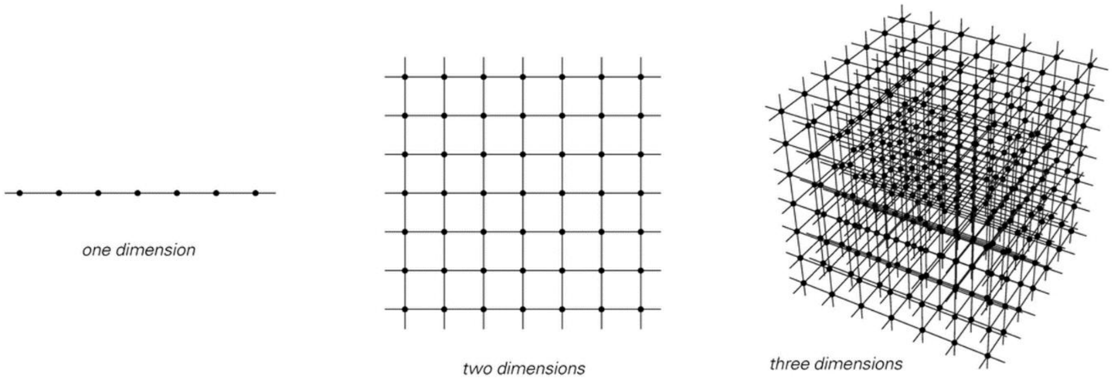


Figure 3.1: Lattice types (Wolfram, 2002).

- Neighborhood: Cells that interact with a given cell within the grid are referred to as its neighbors. A cell may have multiple neighbors, and it responds to predefined rules based on the states of these neighboring cells. There are two primary methods used to define neighborhood relationships: the von Neumann and Moore neighborhoods as shown in Figure 3.2. The von Neumann neighborhood considers the cells located directly above, below, to the left, and to the right of a given cell in a two-dimensional grid. According to this method, each cell has four orthogonal neighbors. The Moore neighborhood includes all eight surrounding cells in a two-dimensional grid. This definition accounts for both orthogonal (top, bottom, left, right) and diagonal (corner) neighbors, asserting that each cell has eight neighboring cells in total.

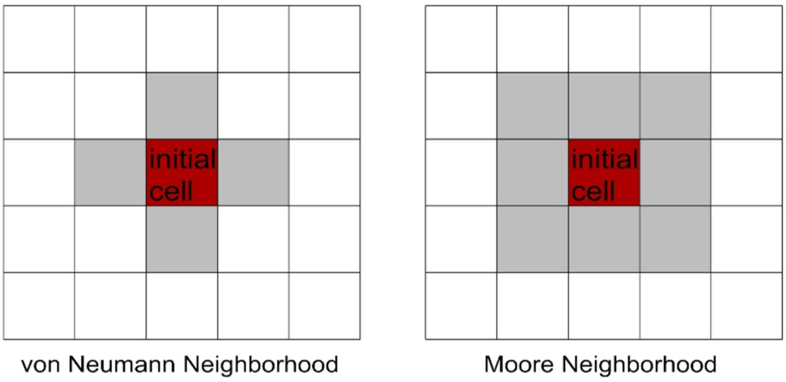


Figure 3.2: Von Neumann and Moore neighborhoods.

- Cell States: In CA, the cell status refers to the discrete state that a cell occupies at any given iteration, which determines its behavior and interaction within the overall system. According to Coates et al. (1996), a CA model can contain multiple cell states beyond the binary alive/dead logic commonly found in simpler models. For instance, in the architectural experiments conducted at the University of East London, cells were assigned states such as 0 for empty, 1 for built volume, and 2 for transitional elements like balconies or external plates. These states are governed by a set of rules—either counting (based on the number of neighbors in a particular state) or voting (influenced by the specific arrangement of surrounding states)—which determine the transition from one state to another. For example, a cell in state 0 (empty) may transition to state 1 if it has a certain number of neighboring cells in state 1, simulating architectural growth. Conversely, transitions to state 2 may occur only under specific spatial constraints, such as being sheltered by built form (Figure 3.3).

Figure 3.3: East London University CA experiments (Coates et al, 1996).

- Time: In CA, time represents discrete steps during which all cells update their states simultaneously based on transition rules. This temporal structure allows complex forms to emerge progressively.
- Transition Rules: In CA, the emergence of new forms depends on a predefined set of transition rules. These rules are typically established by the designer and govern how each cell updates its state in response to the states of its neighbors. Rather than operating independently, all cells apply these rules simultaneously across the grid at each discrete time step. This parallel rule application enables a generative process. The transition rules, therefore, serve as the fundamental logic driving the evolution and formation of spatial or formal structures within the CA system.

According to Wolfram’s (2002) rules, elementary CA uses binary states (0 or 1) for each cell, and their behavior is determined solely by the values of a cell and its immediate left and right neighbors. This means their evolution can be entirely defined by a rule table that maps each of the 8 possible three-cell combinations to an outcome. Since there are 8 such input combinations (2^3), and each can lead to either a 0 or 1, there are 2^8, or 256, unique rule sets. These rules are represented by an 8-bit binary number. When transition rules change, the output differs. There are some examples in Figure 3.4, representing Wolframs’s rule 22, rule 75 and rule 181 outputs.

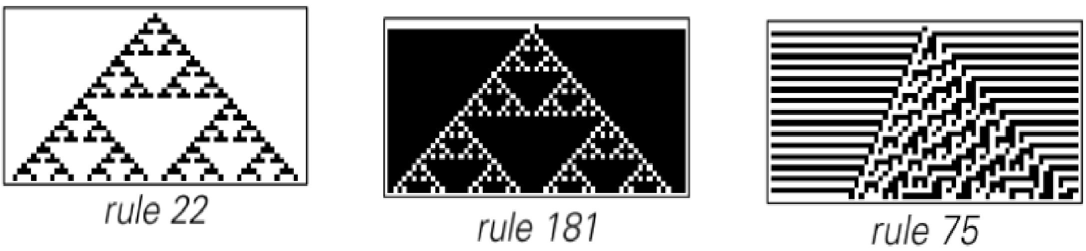


Figure 3.4: Wolfram’s rules (Wolfram, 2002).

One of the most well-known examples of two-dimensional CA is the 'Game of Life' algorithm, developed by John Conway in 1970. The Conway Life Game operates on a two-dimensional grid of squares, where each square can be in one of two possible states: "alive" or "dead." At any given time, a certain portion of the squares in the grid are considered alive (Schulman and Seiden, 1978). Living and dying conditions are determined according to the rules below:

1. A live square will live until the following generation if it has two or three living neighbors; if not, it will die.
2. A dead square will have a "birth" and become alive in the next generation if it has precisely three living neighbors.

All "births" and "deaths" take place simultaneously. While determining the neighbors, the Moore neighborhood is considered.

In Figure 3.5, as illustrated in the preceding example, step 1 involved the cells generating their neighboring counterparts. Subsequently, transitioning from step 2 to step 3, cells with more than three neighbors perished. In the example below, cells with more than three neighbors perished first. Subsequently, additional cells were generated in step 3.

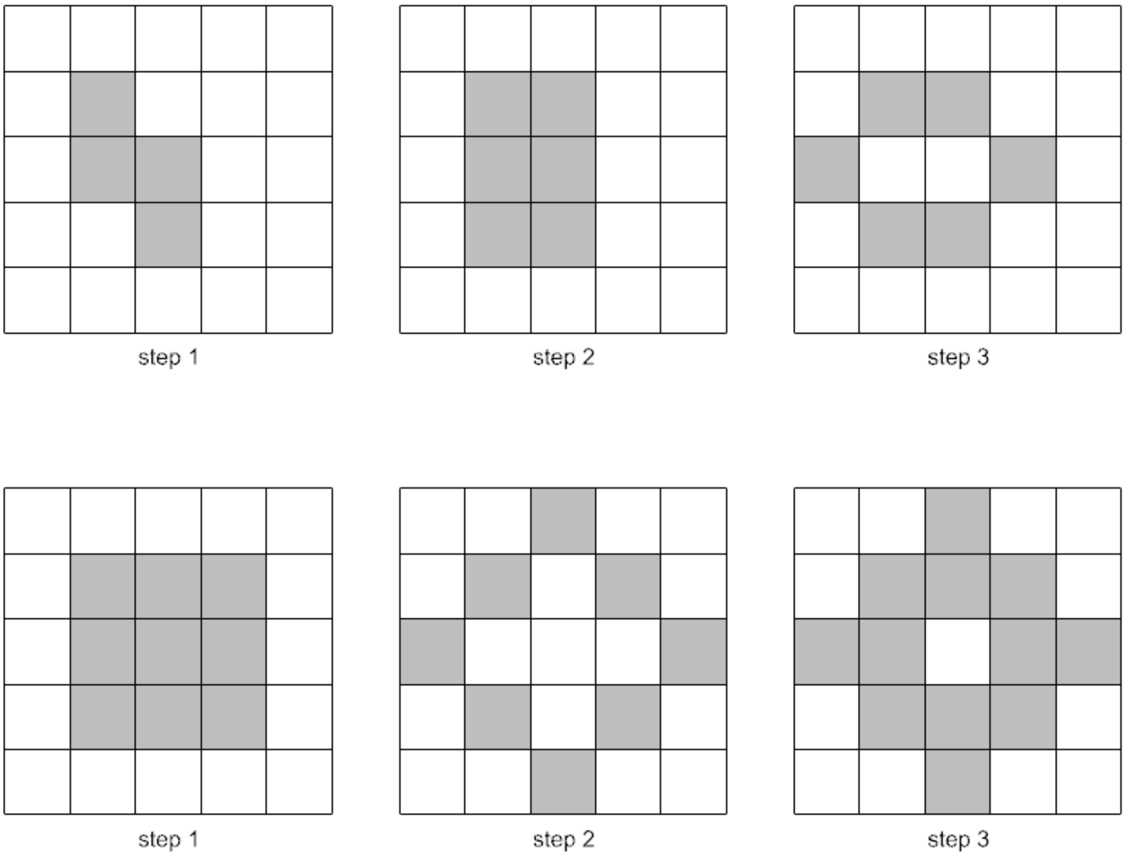


Figure 3.5: Conway’s game of life example.

Moreover, there are examples of CA in 3D. Wolfram (2002), provides examples of 3D CA. In Figure 3.6, the rule dictates that a cell turns black if any of its six face-sharing neighbors were black in the previous step. The system starts with just one black cell.

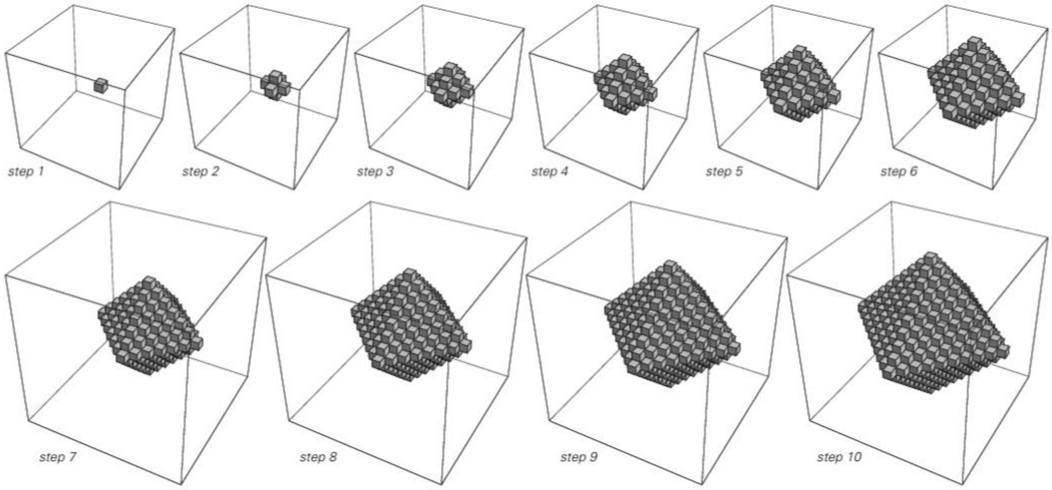


Figure 3.6: Example on 3D cellular automata (Wolfram, 2002).

3.2 Cellular Automata in Architecture

CA systems are particularly encountered in architectural designs that involve modular elements. Since modular systems typically consist of units arranged within a grid and defined by neighborhood relationships, CA-based systems are often employed in such contexts. According to Herr and Kvan (2007), applying CA in architecture necessitates evaluating which aspects of architectural design can effectively benefit from CA, as opposed to those that depend on human judgment. As is common in generative design, the use of CA to automate design processes typically depends on quantifiable design constraints or on structured design methodologies where decisions follow systematic rules.

Moreover, CA models hold the potential to contribute positively to decision-making and can be integrated into various stages of the design process—such as site organization, spatial relationships, circulation, and building envelope design—where interactive, modular, and complex structures are required (Dinçer et al, 2014).

3.2.1 The Generator project

Cedric Price's The Generator project from 1976 was the first time in architecture history that a structure could be thought of as intelligent in both the physical and digital worlds. The initiative was meant to provide a place for Gilman Paper Company workers in Florida to have fun and do things. But it is not at all like an ordinary building in the classic sense. Generator is a system made up of 150 cubes, each measuring 3.6 meters (Figure 3.7). It is completely modular, mobile, and can be reconfigured. These cubes use ordinary industrial parts on concrete bases, and there is no set program or form for the project. The way the structure is laid out can alter all the time depending on what the users are doing, what they like, or even if they aren't interacting with it (Vargas, 2015).

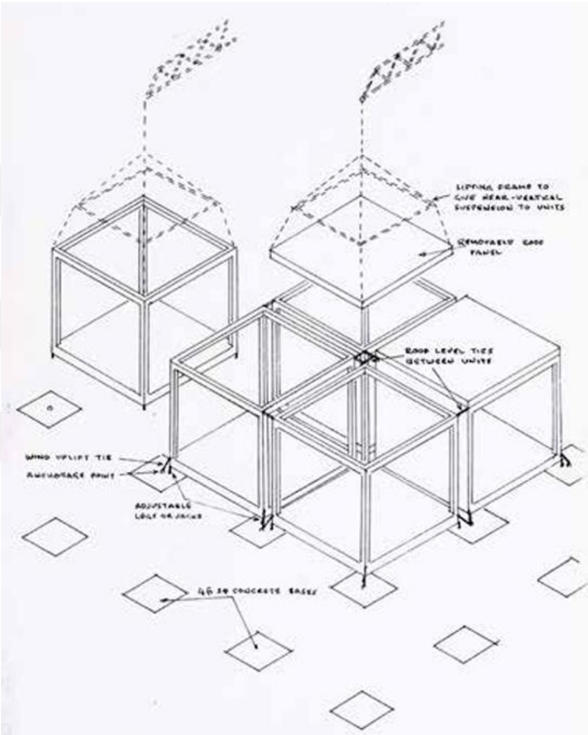


Figure 3.7: Modular nodes in the Generator project (Vargas, 2015).

Although the project does not explicitly mention CA algorithms, several aspects—such as its organization on a grid, the behavior of each modular cube as an individual cell, and the structure's ability to update and reorganize itself over time based on a set of rules—demonstrate clear similarities to CA systems. The project is a concept for a building complex that includes units with varied amenities, such as a guest home, a theater, and a dance studio, all situated within a green space. These functional units are designed to be relocatable over time according to user preferences. Just like in CA, cells that are no longer functional "die" or shift position, while those approved by users "live" or generate new units. Considering all these factors, the project can be regarded as one of the pioneering examples in the history of CA in architecture.

3.2.2 Cellular automata algorithms applied to Parc de la Villette

Devetakovic et al. (2015), searches at how the idea of CA might be used in architecture, using Bernard Tschumi's "folly" buildings at the Parc de la Villette project in Paris as an example. The authors look at how CA could be used in the design of buildings and cities in a theoretical way, using the work of researchers like Krawczyk (2002) and Coates (1996) as examples. The Faculty of Architecture at the University of Belgrade is conducting a series of experimental studies called "Generic Explorations." This research is one of those studies. In this case, architecture students helped with several design trials utilizing the CA module of a software program called Fun3D. The pupils used Tschumi's 3x3x3 cubic cell system as displayed in Figure 3.8 to make new shapes in space by using different CA principles. During the process, a number of different methods were tried out. These included changes to the internal structure of the CA system, variations of a single CA system, and combinations of various CA systems. The new designs keep important aspects of Tschumi's original folly constructions, like size, setting, and color. They also make the idea of cellularity a main focus as a creative design idea. This study shows how CA may be used to create forms and organize space in architectural design. This helps the field of parametric and algorithmic design move forward (Devetakovic et al, 2015).

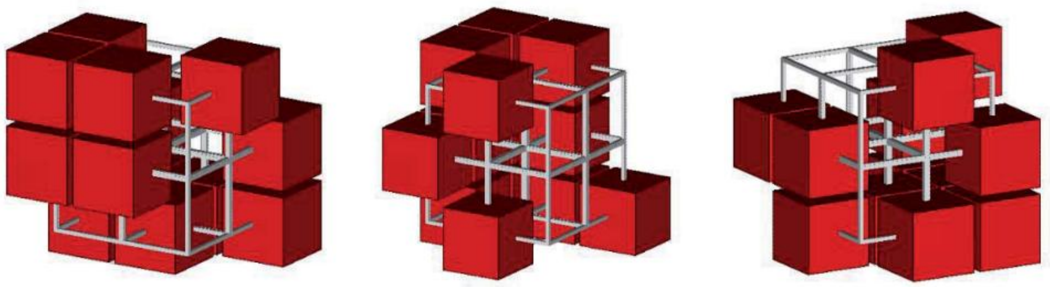


Figure 3.8: 4x4x4 CA modular system for Parc de la Villette (Devetakovic et al., 2015).

3.2.3 Chile Atacama Desert Hotel dwellings

The article of Herr and Ford (2016), presents a comprehensive case study illustrating the adaptation of CA systems as effective instruments in architectural design processes. The project includes the building of a hotel dwelling for scientists and engineers engaged with the European Extremely Large Telescope (E-ELT) in Chile's Atacama Desert. Originally derived from Conway’s Game of Life, the cellular automaton model rapidly showed its insufficiency in meeting the project's particular architectural demands. The designer revised the system by implementing architectural principles that included light, spatial arrangement, structure, scale, and geography. The cell forms and dimensions were modified, particularly to a 3x3x3 meter scale, to provide functional spatial units, and a GIS-based topography model was incorporated to define context-aware rule behaviors. Furthermore, abstract cell states were implemented to indicate architectural characteristics, including public, private, and transitional zones, utilizing color-coded values as seen in Figure 3.9. The CA model transformed from a deterministic shape generator into a conceptual design collaborator through an iterative and interpretive process. The analysis indicates that although CA systems have value in producing formal diversity, their incorporation into architectural design necessitates active interpretation, selective intervention, and contextual adjustment by the designer. The study redefines CA as open-ended conversational tools rather than self-contained automated systems, facilitating reflective design practice through a dynamic interaction between algorithmic generation and human assessment (Herr and Ford, 2016).

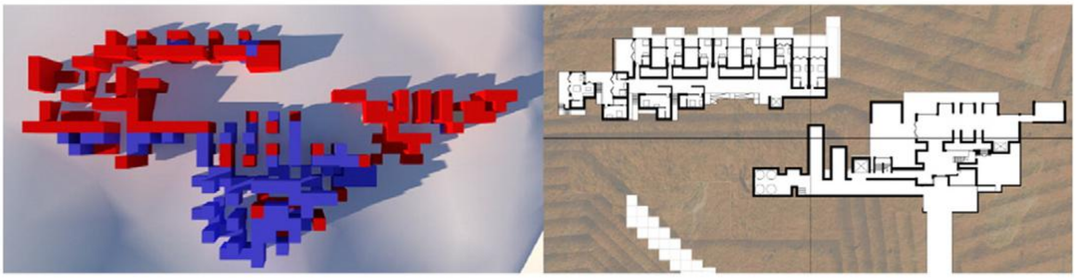


Figure 3.9: The CA model and architectural interpretation (Herr and Ford, 2016).

4. CASE STUDY

This thesis focuses on repurposing the post-disaster container settlements, which were later abandoned when residents moved to permanent housing or left the area. The study explores how participatory architecture and generative design methods can be combined using the MLCA framework. In this section, the applicability of the MLCA algorithm will be presented through a case study focusing on containers left unused after the earthquakes that occurred in Kahramanmaraş on February 6, 2023.

Çalışır Adem and Çağdaş (2022), emphasizes that the CA technique can produce design proposals derived from local urban data, highlighting its potential to adopt a context-sensitive approach. Within the scope of this thesis, the CA model is intended to generate alternatives that are sensitive to the post-disaster context.

4.1 Background

On February 6, 2023, at 04:17 and 13:24, two earthquakes occurred in the Pazarcık and Elbistan districts of Kahramanmaraş, with magnitudes of Mw 7.7 (focal depth = 8.6 km) and Mw 7.6 (focal depth = 7 km), respectively. According to the report by the Presidency of the Republic of Türkiye, Strategy and Budget Directorate (2023), these earthquakes resulted in the loss of life of over 48,000 people, the damage of more than half a million structures, the disruption of communication and energy systems, and substantial financial losses. SBB Post-Seismic Evaluation Report The earthquakes inflicted significant damage on the country, both materially and psychologically, with the most catastrophic consequence being the loss of lives.

As a result of these disasters, thousands of people lost their homes and began living in temporarily established container settlements (Figure 4.1). Since numerous aftershocks continued for a long time following the earthquakes, access to buildings was restricted, and people started meeting all their needs within the container settlements. These settlements serve not only as shelters for local residents but also host facilities that meet their health, food and beverage, and social needs. In the region, 189 container settlements were established, and infrastructure and installation works were carried out for 90,914 containers (Presidency of the Republic of Türkiye, Strategy and Budget Directorate, 2023).

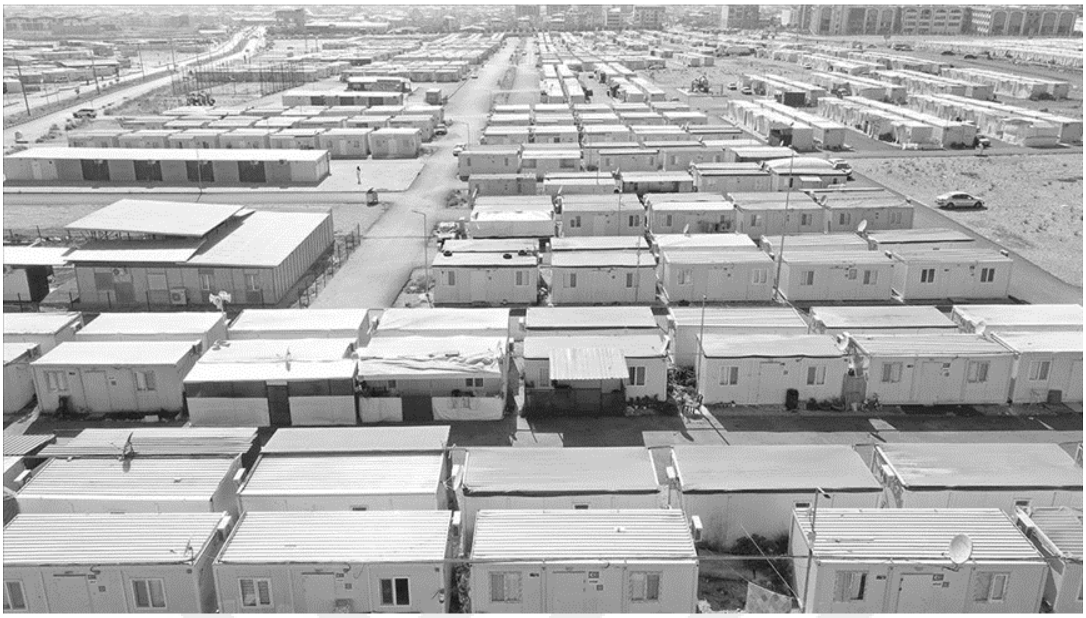


Figure 4.1: One of the container settlements in Kahramanmaraş (URL-4).

These container settlements were established to provide temporary shelter for local residents who had lost their homes. The government initiated new permanent housing projects to revitalize the region and provide living spaces for those affected. As these projects began to be completed as of 2024, residents who received their permanent homes gradually started to leave the container settlements. The evacuation of these container settlements is still ongoing today. However, due to the significant damage not only to superstructures but also to infrastructure caused by the earthquake, some residents are still unable to move into their new homes because of issues such as electricity and water. As a result, container settlements continue to serve as shelter for those who have not yet received permanent housing or who cannot move in due to infrastructure problems. Additionally, beyond their shelter function, container settlements also accommodate damaged social spaces, hospitals, and outpatient clinics.

The evacuated containers are removed from the area without being offered an alternative use and are either sent to the central warehouse or recycled. Considering that significant infrastructure support was provided during the establishment of the container settlements and that a substantial logistical effort was required for their setup, it can be said that the process resulted in financial loss. Similarly, the dismantling process also incurs comparable construction and logistical expenses.

Within the scope of this thesis, the aim is to create alternative settlements and new areas of use for the container units established after the earthquakes in Kahramanmaraş, instead of removing them following the relocation of residents to permanent housing. In forming these living spaces, the goal is to apply collaborative design principles by incorporating the ideas of the earthquake survivors. CA algorithms have been utilized to increase the range of options offered to users and to determine the spatial arrangement of the containers.

4.2 Multi Layer Cellular Automata

While defining settlement rules for MLCA, the aim is to search for functional meaning within the layout proposals generated by the model. According to Kotnik (2017), simultaneously presenting low-dimensional parametric configurations allows designers to take advantage of human perception's natural scanning capacity, making it easier to spot developing spatial effects. As a result, the parametric definition no longer serves solely as an analytical tool, but also as a visual and perceptual design tool, similar to the exploratory possibilities of sketching through ambiguity. This methodology enables for the controlled linking of phenomena such as spatial fluidity, orientation, and light effects to parametric modifications, supporting a design process that combines computational methodologies and intuitive-creative thinking.

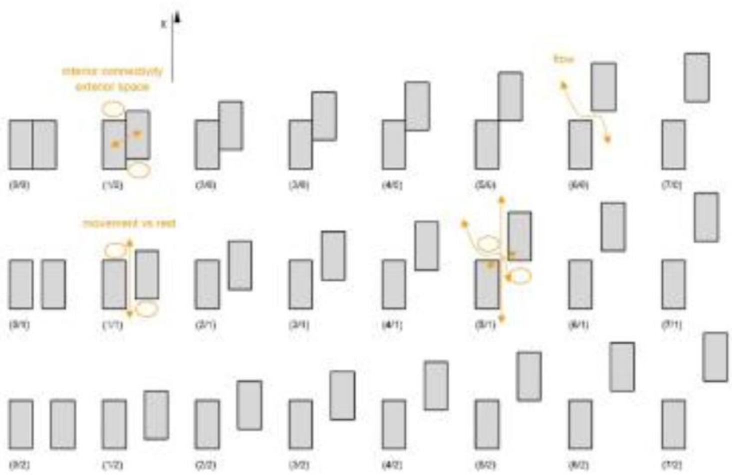


Figure 4.2: Visual identification of spatial phenomena (Kotnik, 2017, p. 42).

The approach discussed by Kotnik in Figure 4.2 is directly relevant to the topic of this thesis. The visual analysis of parametric variations enables designers to intuitively identify spatial phenomena using perception. Similarly, CA algorithms generate variation fields in container settlements, which users can evaluate visually in terms of social space, density, or circulation. As a result, Kotnik's proposed visually and designer-oriented method serves as a key theoretical underpinning for the thesis' goal of merging computational creation with participative, perception-based evaluation.

In the scope of this thesis, "Functional meaning," means that the specific needs of each function are taken into consideration. For example, a food court may require a semi-open space, whereas a market needs to be in a fully enclosed area. On the other hand, for open spaces such as marketplaces or playgrounds, vacant areas must be left within the layout.

Secondly, the parameter of spatial integrity is taken into account in the arrangement of containers. For instance, laundries are spaces that require a more integrated spatial setup, whereas guest houses or social areas are spaces that should have more fragmented massing.

All of the followings are considered as design layers:

Layer 1: Initial Grid
Layer 2: Formal Relations
Layer 3: Functional Relations
Layer 4: Spatial Relations
.
.
.
Layer n : Additional Relations

Each of these layers is provided as an input to the algorithm. Therefore, it can be said that the system operates through a multi layer cellular automata (MLCA) algorithm.

4.3 Grid and Functions

While designing the layout, a grid system with 3m x 3m x 2.5 m cells was created in accordance with CA principles. A standard container, with dimensions of 3m x 6m x 2.5m, was placed onto the grid (Figure 4.3). In other words, a single container consists of two adjacent horizontal cells. Considering that the area is located in an earthquake-prone zone, vertical expansion was avoided, and a maximum of three containers stacked on top of each other was allowed. Therefore, the aim was to create three-story structures with two floors above ground level.

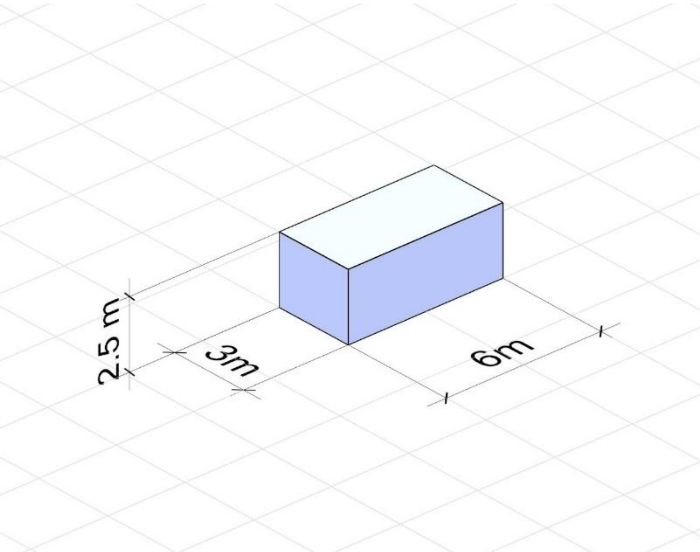


Figure 4.3: Container and grid dimensions.

After determining the standard container dimensions, the functions that might be needed in these settlements following the relocation of container residents to permanent housing were identified.

These functions are determined as follows:

Market
Laundry
Health services
Social Spaces
Library
Guest House
Park
Bazaar Area

When these functions are examined within the scope of MLCA, the spatial area types required by each function were predetermined and incorporated into the planning process. Considering their intended use, some functions require semi-open spaces. The classification of whether a function requires closed, semi-open, or open spaces has been identified and integrated into the algorithm (Table 4.1).

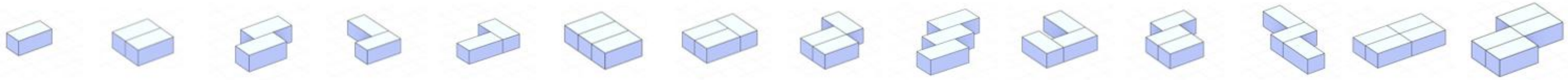
Table 4.1: Functions and enclosure relations.

Floors	Closed Space	Semi-open Space	Open Space
Ground Floor	Laundry, Market	Health services, Social spaces, Libary	Bazaar area, Park
First Floor	-	Library, Guest house, Social spaces	-
Second Floor	-	Guest house	-

It was anticipated that, following the relocation of earthquake victims to permanent housing, the primary function of the container settlement would no longer be to meet basic shelter needs. Guest houses have been included in the container settlement to accommodate individuals who have not yet moved into permanent housing or who, for various reasons, cannot stay in their own homes. These guest houses will serve as the residential units. However, the aim is for the container settlement to evolve into a living center that meets not only shelter needs but also social needs. Therefore, most of the functions assigned to the containers are public spaces. For example, health services will be included to provide space for health services. Additionally, public spaces such as dining areas, libraries, markets, and laundries have been incorporated into the program.

Furthermore, the appropriate floor level for each function has been considered. While classifying the layout, public spaces were placed on the ground floor and first floor, and only guest houses were designated for the second floor. While the settlement was being made, it was taken into consideration to leave space for open areas such as market areas and parks. The mass models were developed by defining whether each programmatic function requires a semi-open or fully enclosed space (Table 4.2). While mass models were being created, it was considered that a maximum 4 containers, 8 cells, could be connected.

Table 4.2: Mass models.

Functions	1 Container Models (2 Cell)	2 Container Models (4 Cell)	3 Container Models (6 Cell)	4 Container Models (8 Cell)
Library				
Social Space				
Health Service				
Market				
Laundry				

For instance, social spaces are required to have semi-open areas. Taking measurements into account, container masses were manually modeled in forms with the potential to create semi-open spaces and tested accordingly. During this testing process, the number of containers needed for each function was also considered. For instance, a space intended for social use can be created with a single 3m x 6m container, while a space such as a library—which needs to have a more cohesive and larger mass—would require the combination of at least three containers. Additionally, when designing the libraries, care was taken to ensure a more cohesive arrangement of the containers.

The whole formed by multiple containers coming together and separated from other container clusters by voids in the design is referred to as a 'module.' In Figure 4.4, the modules arranged within a three-story settlement area are distinguished with different colors.

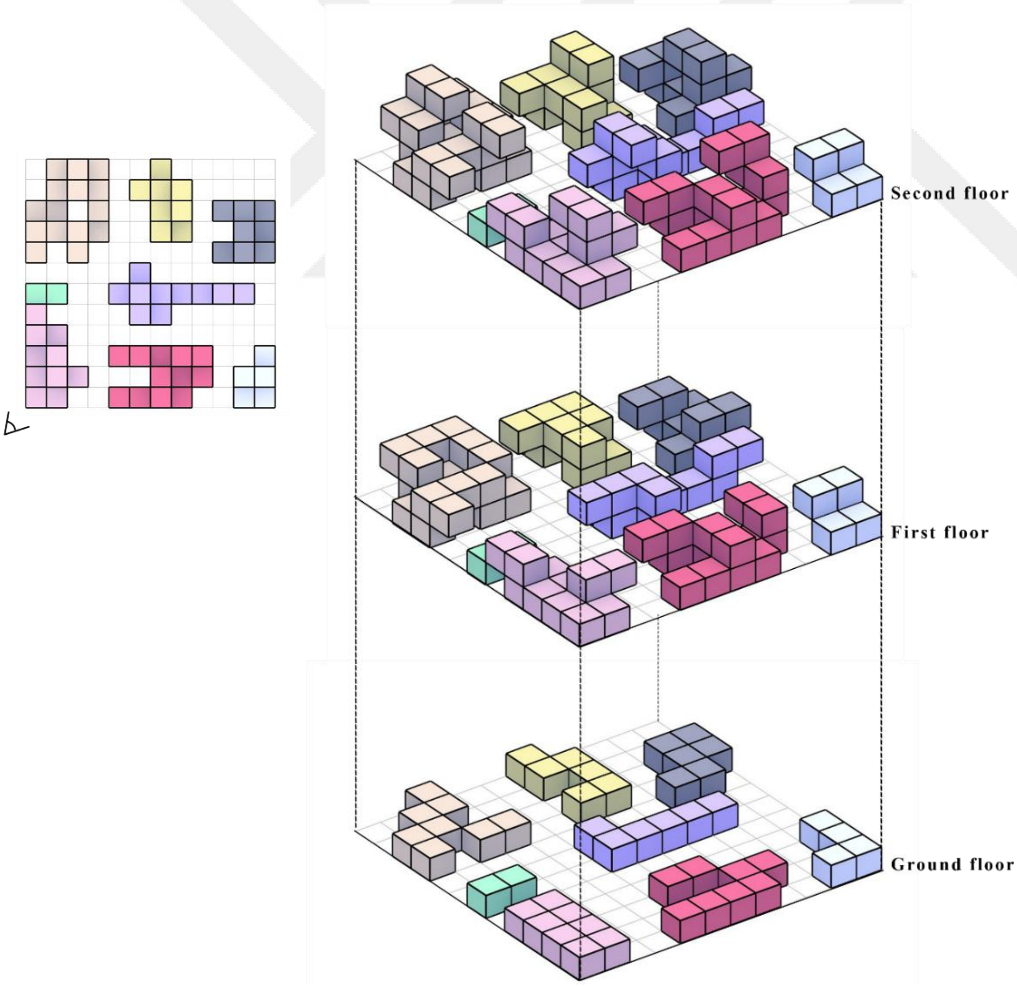


Figure 4.4: Modules.

4.4 Rules of Settlement

Once the container dimensions were defined as 3m x 6m to fit into the 3m x 3m grid system, the next step was to establish the CA rules. The system is designed using the Rhinoceros Grasshopper interface. A Python component is used to integrate the CA algorithm into Grasshopper. In Python script, a grid is first created, and an algorithm is then developed based on the following set of rules to place the containers onto this grid:

1. Cells are placed on a 3m x 3m x 2.5m 3D grid.
2. Each container consists of 2 neighbor cells on the same level (3m x 6m).
3. Since containers are made up of two neighbor cells, an odd number of cells cannot exist on any level—single cells cannot be placed.
4. A maximum of 4 containers can be in contact with each other on the same level.
5. The ground floor (z = 0) is prioritized for container placement. If a container is not placed on the ground floor, no cells can be placed above that area.
6. For containers to be placed on upper levels (z = 1 and z = 2), either the area beneath them must be occupied, or at least one of their adjacent cells in the x or y direction must have a cell beneath it.

After the transition rules are established, parameters that enable generativity are defined by making adjustments within the Grasshopper interface. The first of these parameters are the variables GRID X, GRID Y, and GRID Z, which determine the number of grid lines in the X, Y, and Z planes. Since the goal is to create structures with a maximum of three stories, the GRID Z value is fixed at 3. The GRID X and GRID Y variables can be adjusted according to the dimensions of the site (Figure 4.5).

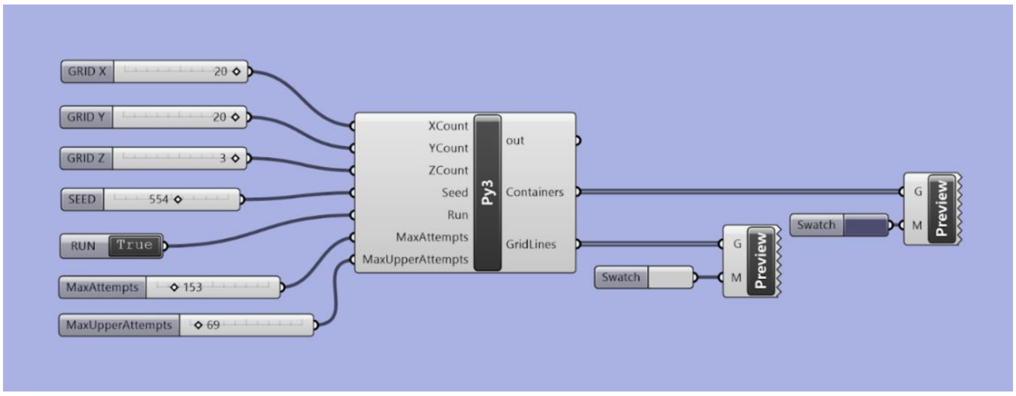
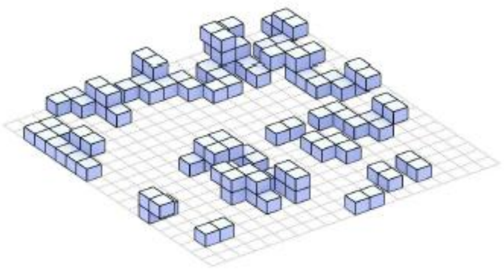
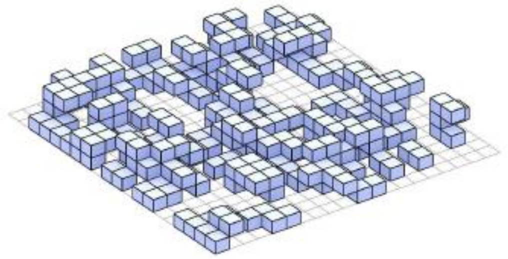
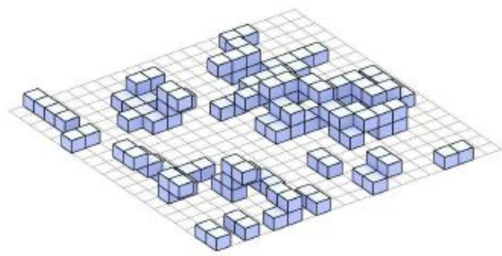
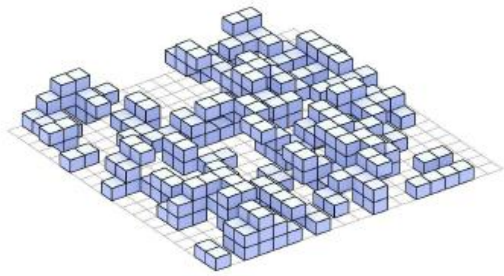
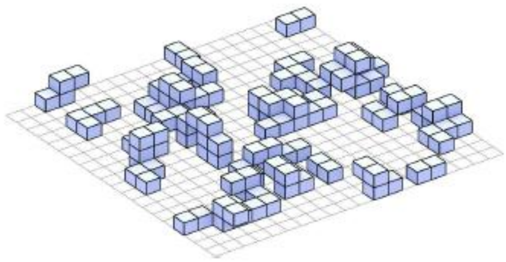
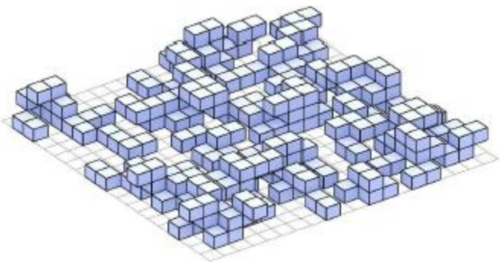
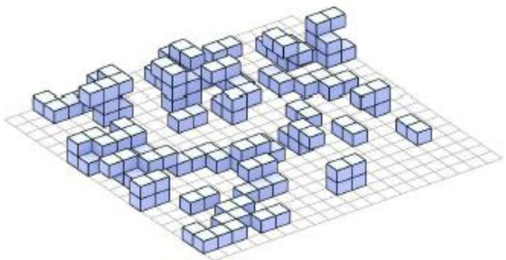
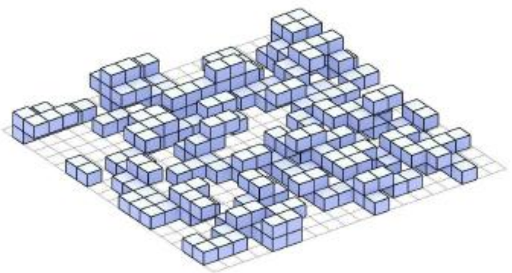


Figure 4.5: Grasshopper components.

The second variable is the SEED parameter. SEED controls the randomness in the placement of containers. In the code, the cells where containers will be placed are selected randomly. However, this randomness produces different results with each run. To enable the observation of different form variations, the SEED variable is connected to a Number Slider in the Grasshopper interface.

The third and fourth variables, MAX ATTEMPTS (MA) and MAX UPPER ATTEMPTS (MUA), control the same parameter for the ground and upper floors, respectively. This parameter defines the number of attempts made to place containers. It limits how many times the algorithm will try to place containers on the upper floors. In each attempt, the algorithm tries to find a valid location for the containers based on the rules; if it fails, it continues trying until the predefined limit is reached. This limit is controlled via Number Sliders added to the Grasshopper interface. In Table 4.3, different settlements can be observed by changing the MA, MUA and SEED parameters.

Table 4.3: Mass generation by changing the parameters.

SEED	MA= 50 MUA=90	MA=120 MUA=150
1		
200		
300		
400		

4.5 Structural Design

Container architecture has recently been utilized for various reasons, such as natural disasters, overpopulation, and limited available settlement areas. The adaptability of containers to mobile and modular design is one of the main reasons for their widespread use. Modularity is based on the principle of advanced standardization. In this context, standardization refers to the establishment of common and repeatable service protocols aimed at addressing specific or potential challenges, thereby ensuring optimized efficiency and order (Sun et al, 2017).

Container buildings are constructed using a modular approach, which differs from conventional structural systems and introduces unique challenges in structural design. According to Su et al. (2022), analysis conducted on a BIM platform with various architectural, mechanical, and energy modeling tools demonstrated that the corrugated steel walls can be integrated with the top and bottom beams. By incorporating diagonal bracing as seen in Figure 4.6, welding the inner metal keel to the corrugated steel plates, and strengthening these connections, the overall structural performance can be significantly enhanced.

Figure 4.6: Diagonal bracing (Su et al., 2022, p. 11).

As a result, the downward deflection of the cantilever structure was effectively minimized, enabling the implementation of a large cantilever design (with a maximum span of 3.6 m) and a spacious layout (Figure 4.7), which successfully met inspection requirements upon completion.

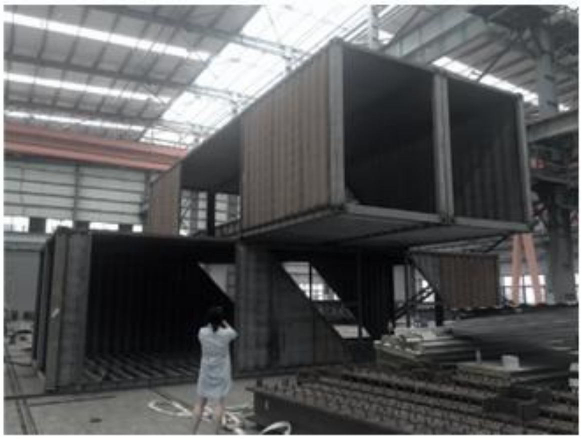
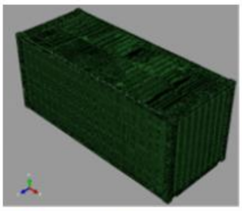
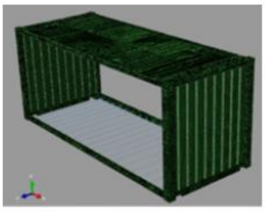
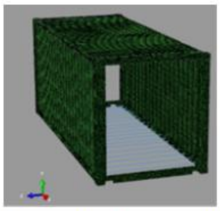
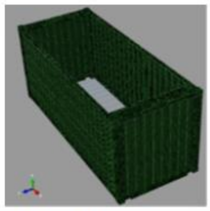
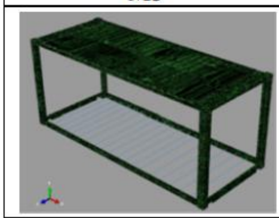
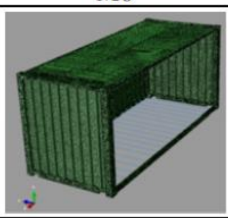
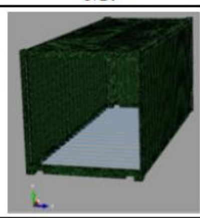
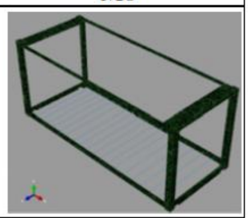


Figure 4.7: Assembly of containers (Su et al., 2022, p. 11).

The workload analyses and loading scenarios conducted by Giriunas et al. (2012) revealed that the main load-bearing elements of shipping containers are the end walls and side walls, while the roof provides almost no structural contribution. End walls are critical under vertical loads, whereas side walls play a key role under longitudinal lateral loads. Five different loading scenarios were applied (Table 4.4); the complete container model (M1) reached the load capacities specified in the ISO standards in all tests, while the modified models generally showed reduced capacity. In particular, the removal of walls significantly decreased stiffness and load-bearing capacity, whereas the removal of the roof did not cause a notable reduction. In conclusion, for containers to be safely used in building applications, end and side walls must be preserved, and connection details must be properly designed.

Table 4.4: Shipping container modifications (Giriunas et al., 2012, p. 53).

M1	M2	M3	M4
			
Original Container	Sidewalls Removed	Endwalls Removed	Roof Removed
M5	**M6**	**M7**	**M8**
			
Sidewalls and Endwalls Removed	Single Sidewall Removed	Single Endwall (Doors) Removed	Sidewalls, Endwalls, and Roof Removed

In the literature, studies on container settlements are predominantly conducted with reference to shipping containers. However, it is important to distinguish between shipping containers, which were originally designed for transport, and housing containers, which are specifically adapted or manufactured for residential use. These two container types may differ in several critical aspects:

- Since shipping containers are used for freight transport, they have a more robust structural system. Housing containers, on the other hand, may be made from modified shipping containers or consist of lighter steel components.
- Housing containers are equipped with systems such as electricity and water installations, whereas shipping containers do not contain these systems.
- Housing containers include thermal insulation layers, while such layers are generally absent in shipping containers.
- In shipping containers, openings such as windows and doors are created later for use, whereas in housing containers production is carried out to include these openings.

In the scope of this thesis, the aim is to create structures in the earthquake-affected area by stacking a maximum of three containers. To ensure the stability and safety of such three-storey settlements, a supporting steel frame structure is required.

This steel frame is not only responsible for carrying the containers but also serves as a system that integrates circulation elements, shading devices, rainwater drainage, and other complementary components. In doing so, it transforms the container clusters into a more complete architectural unit rather than a mere aggregation of boxes.

Regarding the structural elements, I-section steel beams are employed as the main load-bearing members. As illustrated in Figure 4.8, the container walls are positioned directly on top of these beams, allowing loads to be transferred through the frame. Once the container layouts are generated by the algorithm, the structural wireframe is added in alignment with the design, ensuring that the architectural proposal is both spatially functional and structurally feasible.

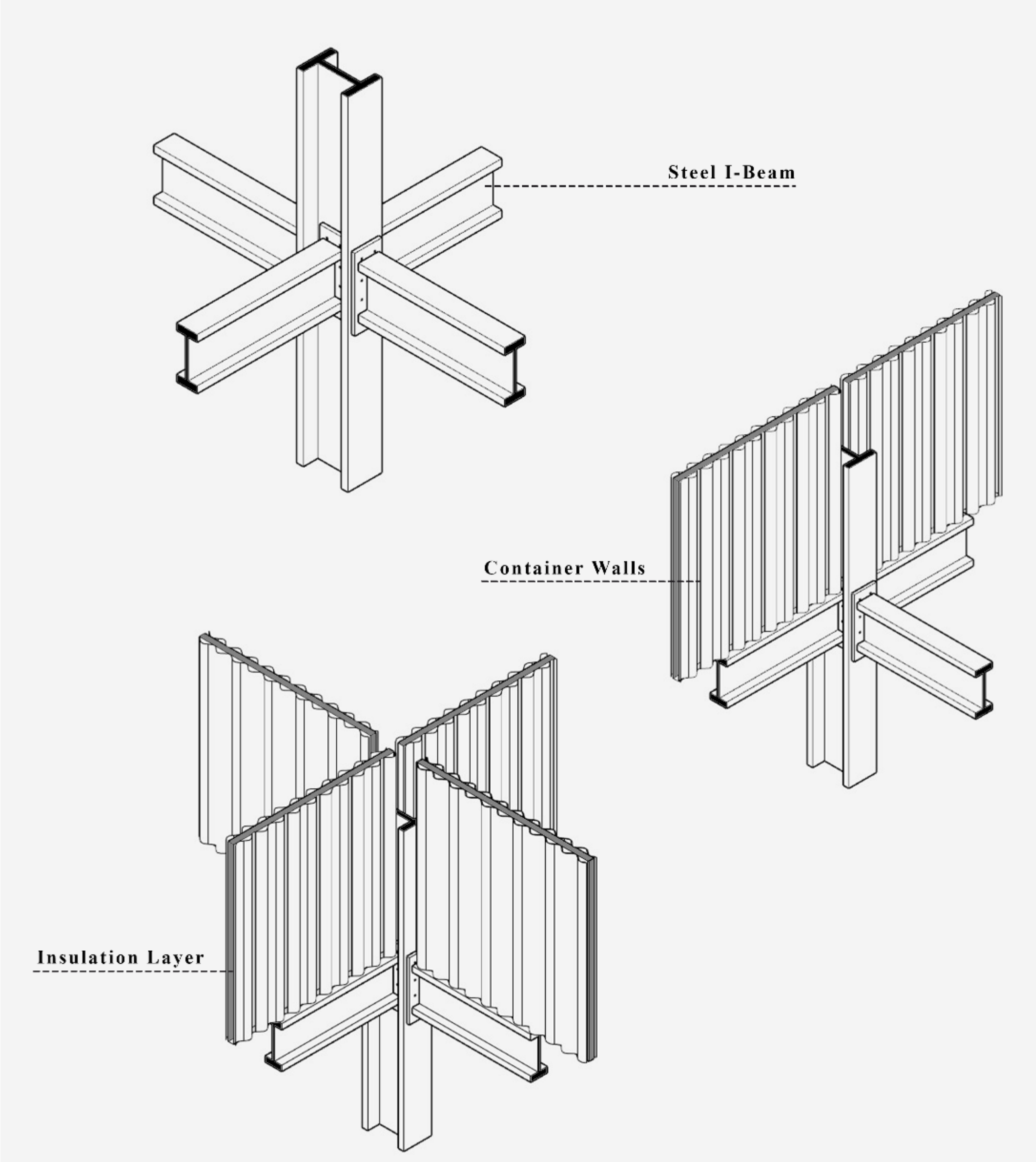
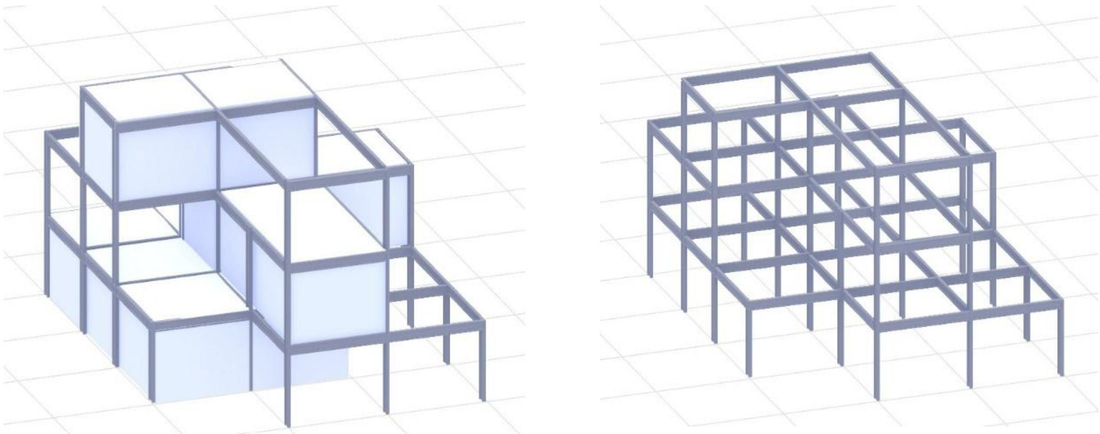
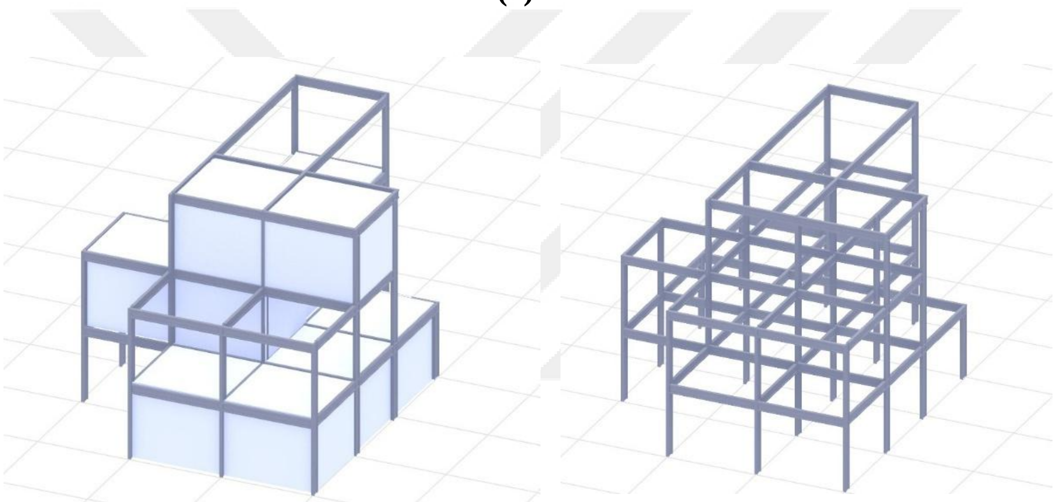


Figure 4.8: Structural details.

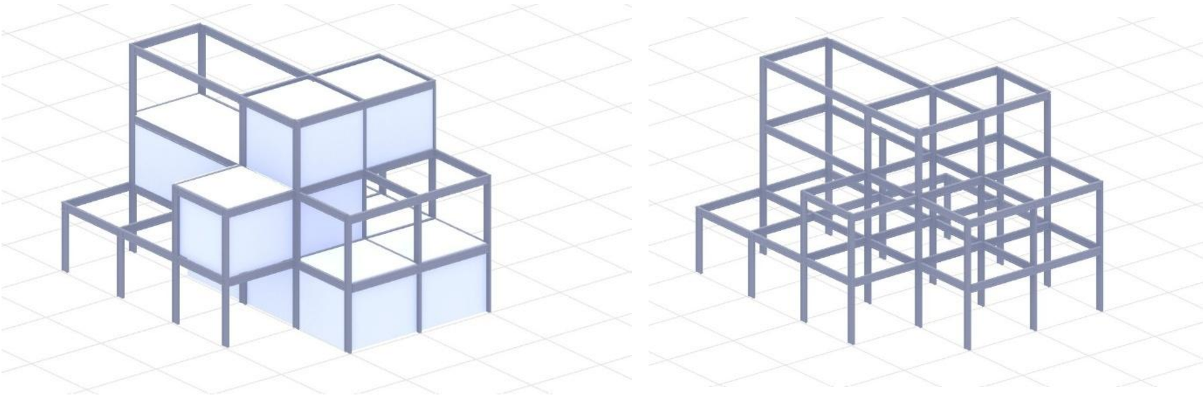
For example, according to rules of settlement, at least one cell of the containers on the first and second floors must be supported underneath. In containers with one supported cell, the other cell is carried as a cantilever. If a circulation element is to be added from the semi-open space created under this cantilever to the upper floor, this element is connected to the steel structure. Likewise, if a terrace is formed on an upper floor, the shading elements belonging to this area are carried by extending the steel structure to cover the terrace. These potentials have been discovered through digital modeling and three of the models can be seen in Figure 4.9.



(a)



(b)



(c)

Figure 4.9: Structural design.

4.6 Circulation

According to the rules of settlement, since a maximum of four containers can be in contact with each other, transition areas such as streets/alleys are included in the models generated by the algorithm in order to ensure open circulation on the ground floor. In addition, if open public spaces (e.g., parks, bazaar areas) are to be added to the design, ground-floor voids can be created by reducing the value of the MA parameter. In this way, it can be observed that open and semi-open public spaces are formed. Containers located above these open spaces, supported by the structural system, create semi-open areas. Figure 4.10 shows an example of an algorithmic design with parameter values of GRID X=12, GRID Y=12, MA=60, and MUA=60.

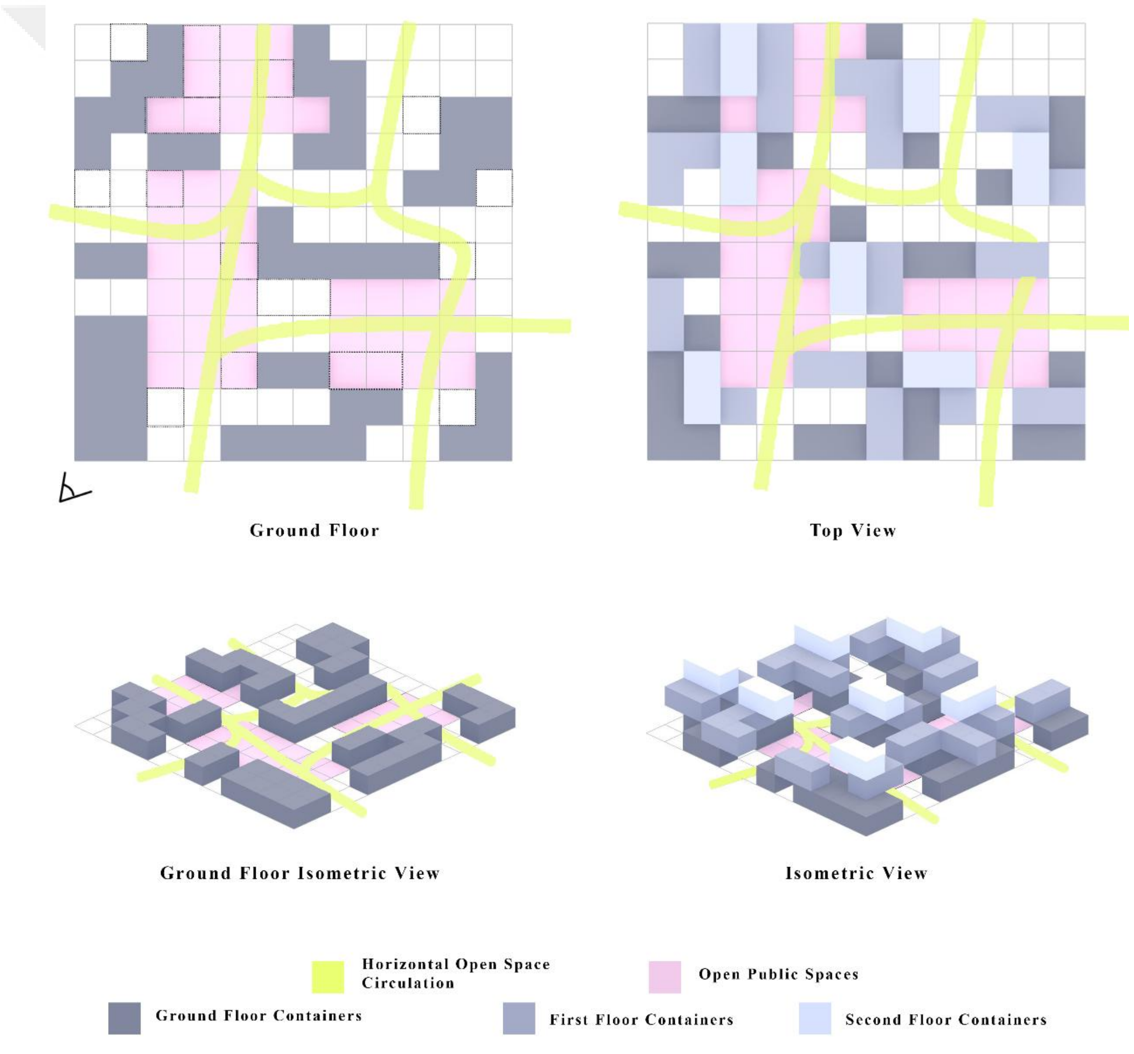


Figure 4.10: Circulation on ground level.

Some functions are planned to extend to the second and third floors. For example, a library function module may expand to the second or third floor. Therefore, internal and external vertical circulation elements are important components in the architectural layout. After the structural system is designed, the placement of external circulation elements is organized. For instance, the semi-open spaces beneath cantilevers or terraces are potential areas where external circulation elements can be located. Alternatively, if a module opens to a public space, vertical circulation elements are added from terraces to this area (Figure 4.11).

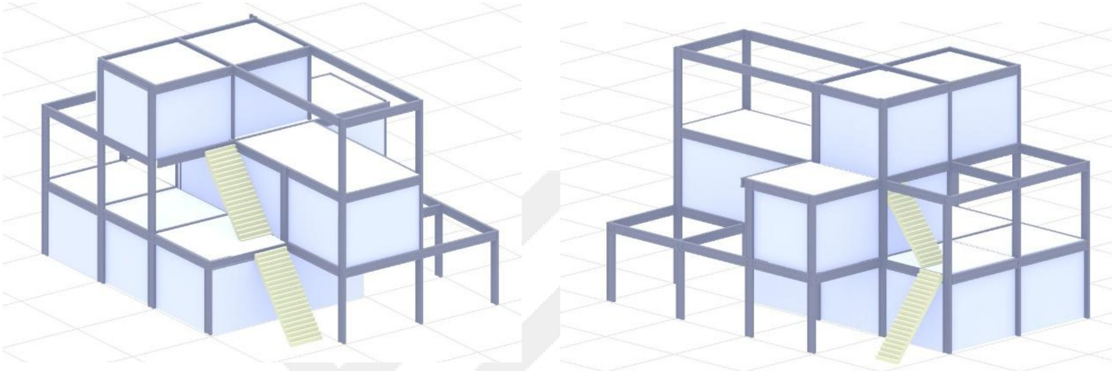


Figure 4.11: Vertical circulation.

4.7 Shading Elements

The extension of the structural system above the semi-open spaces enables the support of additional architectural elements in these areas. Depending on the solar angle and the functional requirements of the intended use, the following three elements as seen in Figure 4.12 and Figure 4.13 may be incorporated:

- Corrugated Metal Sheet: As the primary material of the containers, this element can be sourced from unusable or wasted materials during settlement construction. It may be employed to enclose spaces that must be protected from rainwater and shielded from direct sunlight.
- Perforated Metal Sheet: This element can be utilized in areas where partial permeability to sunlight is desired. For instance, it may serve as a horizontal element over streets and passageways, or as a vertical facade element depending on functional requirements.
- Fabric: At present, fabric shading elements are widely used in container settlements. Due to their air permeability and cost-effectiveness, fabric

membranes are suitable for providing shade over streets and pedestrian corridors during hot summer periods. Moreover, fabrics containing polyurethane offer partial water resistance, which allows them to mitigate the impact of rain. In addition, their flexibility and stretchability enable application in diverse geometries and across multiple structural and floor connections, adding a sense of dynamism to the overall design.

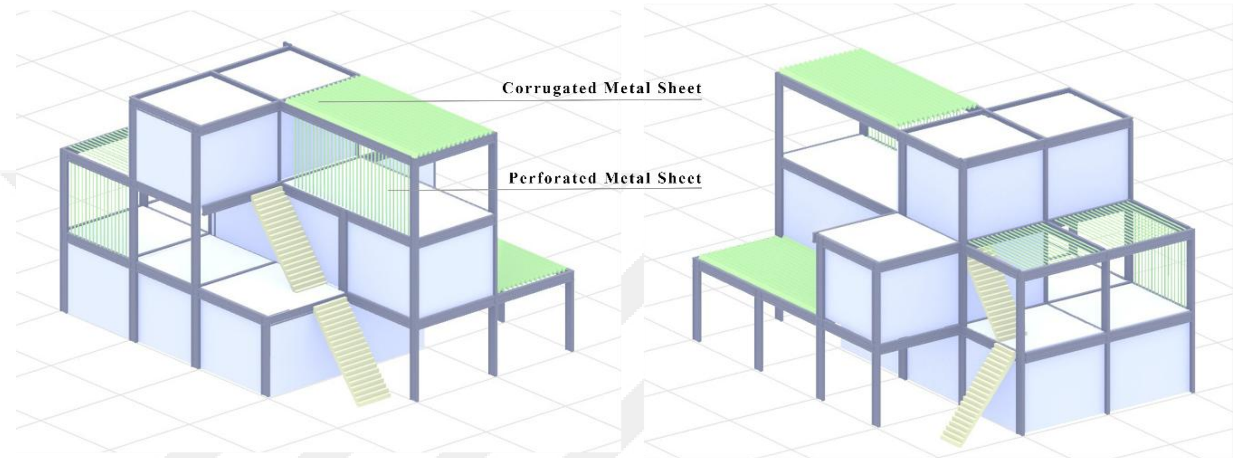


Figure 4.12: Shading elements.

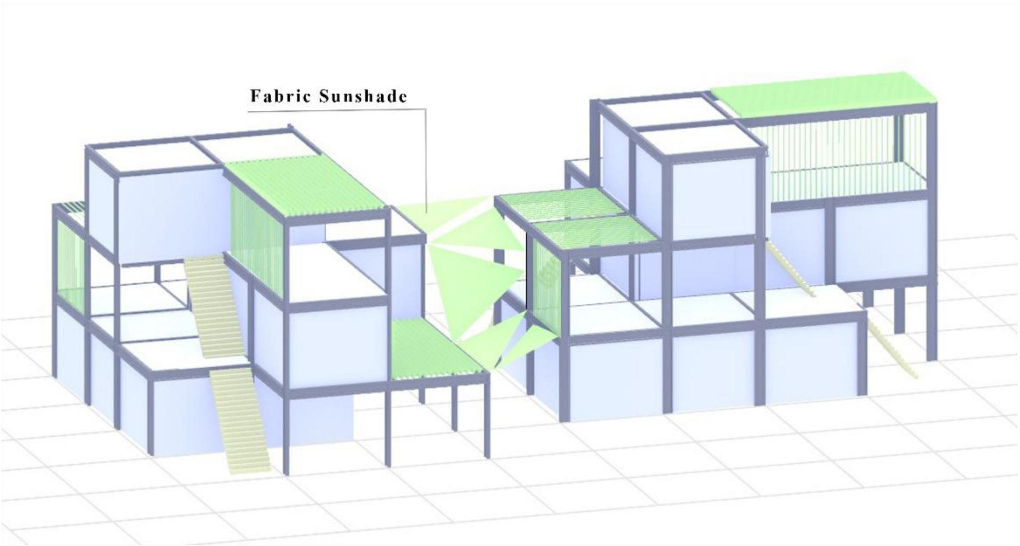


Figure 4.13: Fabric shading elements in triangulated form.

In summary, the design process that began with the removal of containers from the area is outlined below in steps. Moreover, the stages and design elements can be further explored in the exploded view in Figure 4.14.

1. First, the site boundaries are defined, and a grid compatible with the CA algorithm is selected.
2. Considering the desired densities on the ground and upper levels, parameters are entered, and the mass model is generated.
3. Once the mass model is produced, architectural functions predetermined in relation to spatial uses (open/closed spaces) are sought within the modules. Functions are assigned to appropriate modules, while the algorithm is rerun for unsuitable areas.
4. After the mass placements are completed, the structural system is designed. At this stage, shading elements are also taken into account, and necessary extensions are planned accordingly.
5. Interior and exterior circulation elements are designed. On the ground floor, pathways leading to public spaces and the accessibility of such spaces are considered. For the upper levels, exterior circulation elements are integrated by utilizing semi-open and open spaces.
6. Shading elements are positioned in accordance with the structural system. At this stage, the degree of protection—whether full shading or partial permeability—is determined based on the designated function of the container, and the appropriate shading element is selected. Also the external circulation areas are covered with fabric shading elements if necessary.

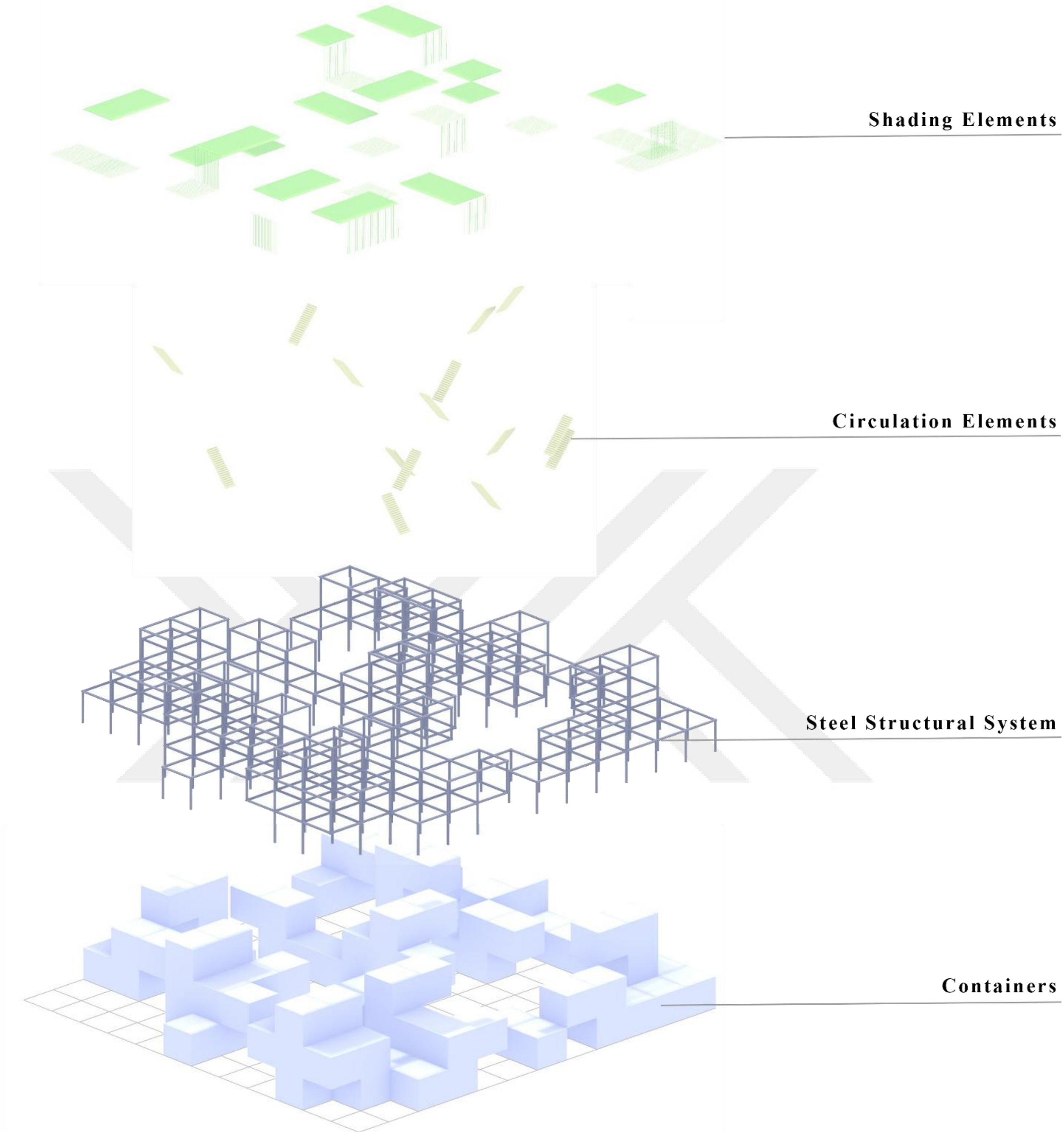


Figure 4.14: Exploded isometric view of design stages.

4.8 User Interaction and Graphical User Interface

The participatory perspective is introduced to the project at the initial stages. The participant, who does not possess technical knowledge, uses the interface designed specifically for them to adjust the parameters GRID X, GRID Y, GRID Z, MA, MUA, and SEED according to their needs and preferences, and then selects the design most suitable for them. In this process, it is assumed that the participant has no technical background, so the parameters are expressed in simplified terms. For example, the parameters GRID X, GRID Y, and GRID Z are controlled through the sliders SQUARED ALONG WIDTH, SQUARED ALONG DEPTH, and FLOORS. The parameters MA and MUA are linked to the sliders GROUND FLOOR DENSITY and UPPER FLOORS DENSITY. The CHANGE slider controls the SEED value and can also be activated by pressing the R key on the keyboard to regenerate the model (Figure 4.15). For model visualization and camera navigation, the W and S keys adjust the model angle, the LEFT/RIGHT keys rotate the model horizontally, and the UP/DOWN keys control zoom in and zoom out. The principles of simplicity and clarity have been prioritized in the design and use of the GUI.

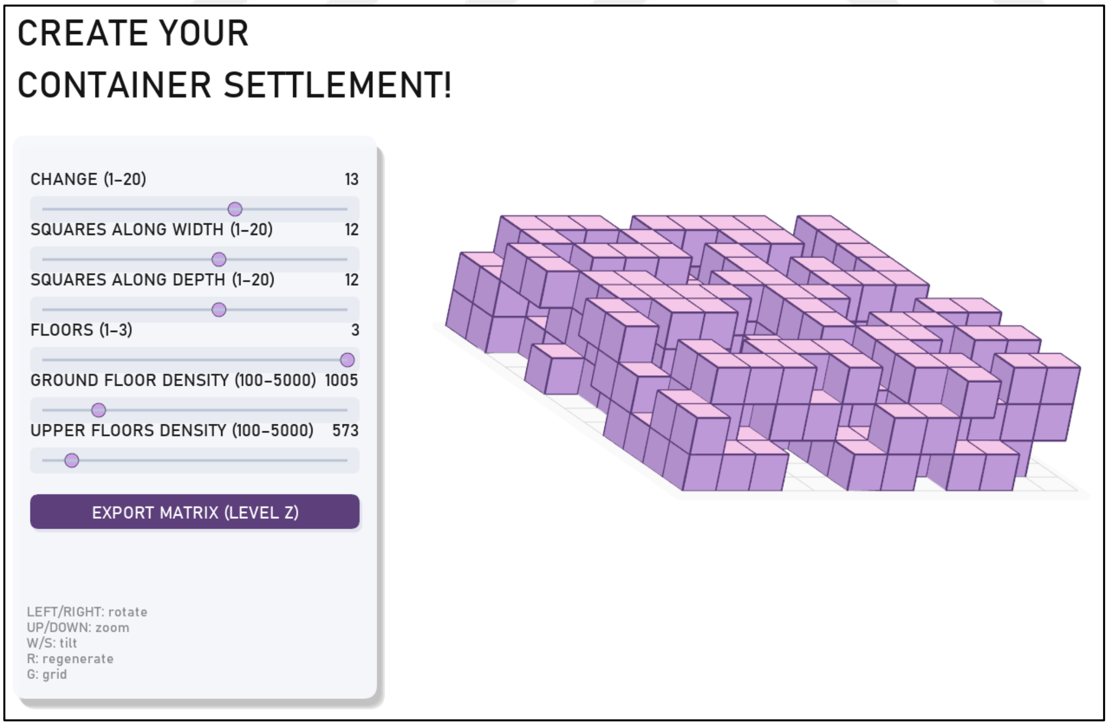


Figure 4.15: GUI.

For the model generation, the Processing software based on the Java programming language was used. The code originally written in Python within Grasshopper was translated into Java at this stage, and the scripts providing the GUI features mentioned above were integrated into the code. After the users makes their selection, the EXPORT MATRIX option is clicked. This option exports the modeled container layouts through a matrix to the designer. For example, when an export is taken for a 5x5 grid with three floors, the model shown in Figure 4.16 is obtained.

LEVEL Z=0	LEVEL Z=1	LEVEL Z=2
0;1;1;0;0	0;0;0;0;0	0;1;0;1;0
0;1;1;1;0	0;1;0;1;0	0;1;0;1;0
0;1;1;0;0	0;1;0;1;0	1;1;1;0;0
1;0;1;0;0	1;1;1;1;0	1;0;1;0;0
1;0;0;1;1	0;0;0;0;0	0;0;1;0;0

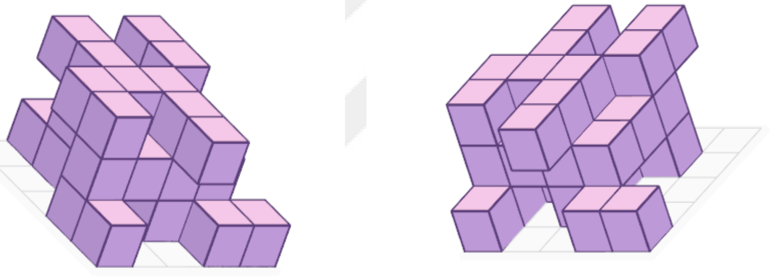


Figure 4.16: 5x5x3 generation example.

In conclusion, the GUI allows participants to incorporate their preferences and needs into the design, functioning as a proof of concept that the CA algorithm is capable of producing alternative living environments adapted to the local context.

5. CONCLUSIONS AND RECOMMENDATIONS

This thesis aimed to reevaluate container settlements that were established after the 2023 Kahramanmaraş earthquakes and later abandoned after their residents were either relocated to permanent housing or left the area. In this context, the study sought to integrate participatory architecture and generative design methodologies through MLCA. The spatial and social incompatibilities of post-disaster temporary settlements necessitated questioning the potential transformation of these areas into more economically, socially, and environmentally sustainable phases for long-term use. Accordingly, the study proposed a design approach that not only responds to functional needs but also incorporates social requirements through community participation.

The main objective of the study was to establish a system that could generate meaningful spatial configurations by contextualizing the CA algorithm within an architectural framework while addressing functional requirements. What distinguishes this work from others is its effort to derive meaning from the design alternatives produced by CA and the incorporation of a functional layer into a MLCA algorithm. Spatial arrangements based on functional needs—such as healthcare units, libraries, dining halls, guest houses, and open/semi-open social areas—were modeled through CA rules. The algorithm, developed in Rhino-Grasshopper using Python, simulated container placements on a modular 3D grid of 3x3x2.5 meters, in accordance with stacking and horizontal connectivity rules. Earthquake risk and functional requirements of the site were also integrated into these rules.

MLCA system generated various settlement scenarios by considering the formation of open and semi-open spaces. The ability of the system to generate meaningful outputs through layers such as function and collaboration demonstrates that generative design tools can go beyond pure form-making to produce socially and contextually meaningful spaces. One of the goals of the study is to evaluate these generated scenarios to establish a form-function relationship.

One of the most original aspects of this study is the addition of a functional meaning layer to the generative CA algorithm. Thus, algorithmic outputs were transformed from arbitrary forms into meaningful life scenarios tailored to human experience. Additionally, relationships between different functions were analyzed through this semantic layer. In this respect, MLCA has the potential to generate alternative living spaces for post-disaster, abandoned container settlements. The fact that the produced configurations include not only formal aspects but also social spaces such as transition zones, courtyards, and semi-open areas that offer livable environments for users positions the system beyond a mere algorithmic solution—toward one capable of producing sustainable living environments in social, economic, and environmental terms.

The model, coded in Python and presented through Grasshopper and Rhinoceros, can serve for designers who wish to employ MLCA in their projects. Since the algorithm can be executed and modified through script adjustments and the use of program tools, it directly addresses experts. As the developed algorithm provides an estimation of the potential outcomes of early design decisions, it can be utilized as a decision support system. On the other hand, the interface created in Java via Processing has been articulated in the language of non-expert users. Within this interface, users are able to visualize the results of their own decisions; therefore, it is intended to serve not only professionals but also users and participats of the project without technical expertise.

From a participatory architecture perspective, user evaluation of these alternatives by a GUI add a collaborative dimension to the study. Thus, the user has been enabled to determine the choices in line with their own desires and preferences through ana GUI that is simple and easy to use, without the need for technical knowledge, and to observe the results. With the commencement of construction of the project approved by the users, it has been intended that the local community also plays a major role in the decision-making process. As a result, the GUI, which enables participants to reflect their choices and needs in the design, serves as a proof of concept demonstrating that the CA algorithm can generate alternative living spaces tailored to the local context.

5.1 Recommendations for Further Studies

In the current state, the analysis of which container may accommodate which function within the mass models generated by the algorithm is carried out by the architectural designer. In future studies, function inputs can be integrated into the algorithm as transition rules. Accordingly, the automated process referred to in Figure 1.1 can potentially be further enhanced. For instance, if a module is to be designed to include social spaces, when the algorithm encounters the social space forms listed in Table 4.2 during its iterations, those forms can be fixed, and the process can continue with the neighboring cells. In this case, instead of searching for architectural meaning within the model generated by the MLCA, the function input will have been pre-defined for the algorithm, thereby allowing the produced model to be more controlled. Furthermore, it will also be possible to define which functions may exist within neighbor relationships, as well as their proximity/distance relationships to one another

The use of predefined forms generated by the algorithm may also enable further advancements in the GUI. For example, when a container group consisting of three containers is clicked through the GUI, the user may be presented with suggestions regarding which function this group was created for and which other functions it may be suitable for. Additionally, by displaying which functions are present in the neighboring containers, the decision-making process can be further supported.

In conclusion, at present the MLCA generates mass models, implemented through Python and Java codes and other digital design tools, that propose spatial arrangements by considering transition rules. However, in the produced models, no form–function relationship has been defined within the coding. This connection is established through the integration of the 'Functional Relations' and 'Spatial Relations' layers into the MLCA algorithm by the designer. In this framework, a design system open to the designer’s interpretation and intervention has been established. By incorporating these two layers into the coding, the generated models could ultimately become the final output of a fully automated process.

REFERENCES

Afet ve Acil Durum Yönetimi Başkanlığı (AFAD). (2023). 06 Şubat 2023 Pazarcik-Elbistan Kahramanmaraş (Mw: 7.7 – Mw: 7.6) Depremleri Raporu. Ankara: AFAD.

Aravena, A., & Iacobelli, A. (2016). Elemental: Incremental housing and participatory design manual. Ostfildern: Hatje Cantz.

Coates, P., Healy, N., Lamb, C. and Voon, W.L. (1996). The Use of Cellular Automata to Explore Bottom Up Architectonic Rules. Eurographics Conference. Imperial College of Science and Technology, London.

Çalışır Adem, P., & Çağdaş, G. (2022). Cellular automata for infill designs in historic urban quarters. Nexus Network Journal, 24(3), 567–586. https://doi.org/10.1007/s00004-022-00602-2

Çelik, S. D. (2022). Participatory design process in post-disaster housing production: The case of October 30, 2020 Izmir earthquake (Master's thesis, İzmir Institute of Technology). İzmir Institute of Technology Repository.

Devetakovic, M., Petrusevski, L., Dabic, M. ve Mitrovic, B. (2009). Les Folies Cellulaires – An Exploration in Architectural Design Using Cellular Automata. 12th Generative Art Conference 2009.

Dinçer, A. E., Çağdaş, G., & Tong, H. (2014). A computational model for mass housing design as a decision-support tool. Procedia Environmental Sciences, 22, 270–279. https://doi.org/10.1016/j.proenv.2014.11.026

Duarte, J. P. (2001). Customizing mass housing: a discursive grammar for Siza's Malagueira houses. Mit. http://papers.cumincad.org/cgi-bin/works/Show?diss_duarte.

Giriunas, K., Sezen, H., & Dupaix, R. B. (2012). Evaluation, modeling, and analysis of shipping container building structures. Engineering Structures, 43, 48–57.

Gu, N., Singh, V., & Merrick, K. (2010). A framework to integrate generative design techniques for enhancing design automation. Proceedings of the 15th International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA), 127–136.

Hernández Vargas, J. (2015). From the Fun Palace to the Generator: Cedric Price and the conception of the first intelligent building. ARQ (Santiago. Impresa), 90, 48–57. https://doi.org/10.4067/S0717-69962015000200007

Herr, C. M., & Ford, R. C. (2016). Cellular automata in architectural design: From generic systems to specific design tools. Automation in Construction, 72, 39–45. https://doi.org/10.1016/j.autcon.2016.07.005

Herr, C. M., & Kvan, T. (2007). Adapting cellular automata to support the architectural design process. Automation in Construction, 16(1), 61–69. https://doi.org/10.1016/j.autcon.2005.10.005

Jiao, J. & Boerboom, L. (2006). Transition Rule Elicitation Methods for Urban Cellular Automata Models. In J. P. v. Leeuwen, H. J. P. Timmermans (Eds) Innovations in Design&Decision Support Systems in Architecture and Urban Planning. Netherlands: Springer, 53-68.

Koçer Özgün, F. N., & Alaçam, S. (2023). An evaluation of augmented reality–based user interfaces in the design process. Architecture and Planning Journal (APJ), 28(3), Article 39. https://doi.org/10.54729/2789-8547.1234

Kotnik, T. (2017). On the Role of Geometry in Formal Design. Architectural Research in Finland, 1(1), 38–47.

Krawczyk, R. J. (2002). Architectural Interpretation Of Cellular Automata. International Conference On Generative Art 2002.

Lefebvre, H. (2003). The Urban Revolution (R. BONONNO, Trans.). University of Minnesota Press. (Original work published 1970)

Lorenzon M. (2023). Earthen Architecture as a Community of Practice: A Case Study of Neolithic Earthen Production in the Eastern Mediterranean. Cambridge Archaeological Journal. 2023;33(4):601-618. doi:10.1017/S0959774323000033

Luck, R. (2018). Participatory design in architectural practice: Changing practices in future making in uncertain times. Design Studies, 59, 139–157. https://doi.org/10.1016/j.destud.2018.10.003

Makadi, Y. C., Arlikatti, S., Zewdu, D., & Maghelal, P. (2025). Review of Temporary Shelter Planning Models: Global Trends and Evidence from Ongoing Practices. Natural Hazards Review, 26*(4).* https://doi.org/10.1061/NHREFO.NHENG-2339

O'Brien, D., Carrasco, S., & Dovey, K. (2020). Incremental housing: harnessing informality at Villa Verde. Archnet-IJAR, 14(3), 345-358. https://doi.org/10.1108/ARCH-10-2019-0237

Presidency of the Republic of Türkiye, Strategy and Budget Directorate. (2023). 2023 Kahramanmaraş and Hatay earthquakes report. *https://www.sbb.gov.tr/wp-content/uploads/2023/03/2023-Kahramanmaras-and-Hatay-Earthquakes-Report.pdf*

Reed, S., Hsi, S., Kreylos, O., Yikilmaz, M.B., Kellogg, L.H., Schladow, S.G., Segale, H., and Chan, L. (2016). Augmented Reality Turns a Sandbox into a Geoscience Lesson, EOS 97, https://doi.org/10.1029/2016EO056135

Rezoug, A. (2013). Fast, cheap & adaptable: A digital model for designing temporary post-disaster housing (Master's thesis). İstanbul Teknik Üniversitesi.

Schulman, L. S., & Seiden, P. E. (1978). Statistical mechanics of a dynamical system based on Conway's game of Life. Journal of Statistical Physics, 19(4), 293–314. https://doi.org/10.1007/BF01011727

Su, Y., Guo, S., Yang, J., & Zhang, Z. (2022). Research on integrated design of modular steel structure container buildings based on BIM. Advances in Civil Engineering, 2022, 1–15. https://doi.org/10.1155/2022/1566284

Sun, Z., Mei, H., & Ni, R. (2017). Overview of modular design strategy of the shipping container architecture in cold regions. IOP Conference Series: Earth and Environmental Science, 63*(1), 012035.* https://doi.org/10.1088/1755-1315/63/1/012035

Şen-Bayram, A. K., Torus, B., Doma, O. O. & Şener, S. M. (2023). Procedural architectural settlement generator for container housing: A study on Marmara and Mediterranean Regions. Estoa. Revista de la Facultad de Arquitectura y Urbanismo de la Universidad de Cuenca, 12(24), 161-173. https://doi.org/10.18537/est.v012.n024.a13

Tan, E., & Portugali, J. (2012). The Responsive City Design Game. In J. Portugali, H. Meyer, E. Stolk, & E. Tan (Eds.), Complexity Theories of Cities Have Come of Age: An Overview with Implications to Urban Planning and Design (pp. 369–390). Springer. https://doi.org/10.1007/978-3-642-24544-2_18

Terlemez, A., K. (2018). Uygulamalı Katılımcı Mimarlığın Türkiye'deki Bağımsız Mimari Gruplar Üzerinden İncelenmesi. The Turkish Online Journal of Design, Art and Communication. January 2018 Volume 8 Issue 1, pp.143-152.

Vardouli, T. (2012). Design-for-empowerment-for-design: computational structures for design democratization. https://dspace.mit.edu/handle/1721.1/72864

von Neumann, J. (1966). Theory of self-reproducing automata (A. W. Burks, Ed.). University of Illinois Press.

Wolfram., S. (2002). A New Kind of Science. Champaign: Wolfram Media.

URL-1 <*https://duzceumutatolyesi.wordpress.com/5n-1k/*> date retrieved 29.05.2025

URL-2 <*https://www.archdaily.com/780544/three-years-in-villa-verde-elementals-incremental-housing-project-in-constitucion-chile*> date retrieved 29.05.2025

URL-3 <*https://web.cs.ucdavis.edu/~okreylos/ResDev/SARndbox/*> date retrieved 29.05.2025

URL-4 <*https://www.aa.com.tr/tr/gundem/depremlerin-merkez-ussunde-4-konteyner-kent-tahliye-edildi/3302449#*> date retrieved 29.05.2025

URL-5 <*https://udcsa.gsd.harvard.edu/projects/3*> date retrieved 01.09.2025

URL-6 <*https://time.com/6256495/rebuild-pakistan-floods-architect-yasmeen-lari/*> date retrieved 13.09.2025

URL-7 <*https://www.intbau.org/climate-volunteers-in-pakistan-workshops-with-yasmeen-lari/*> date retrieved 13.09.2025

URL-8 *<https://www.dezeen.com/2023/05/03/dezeen-in-depth-yasmeen-lari-architects-role-more-activist/*> date retrieved 13.09.2025

APPENDICES

APPENDIX A: Grasshopper Components
APPENDIX B: Python Script
APPENDIX C: Java Script
APPENDIX D: Renders

APPENDIX A: Grasshopper Components

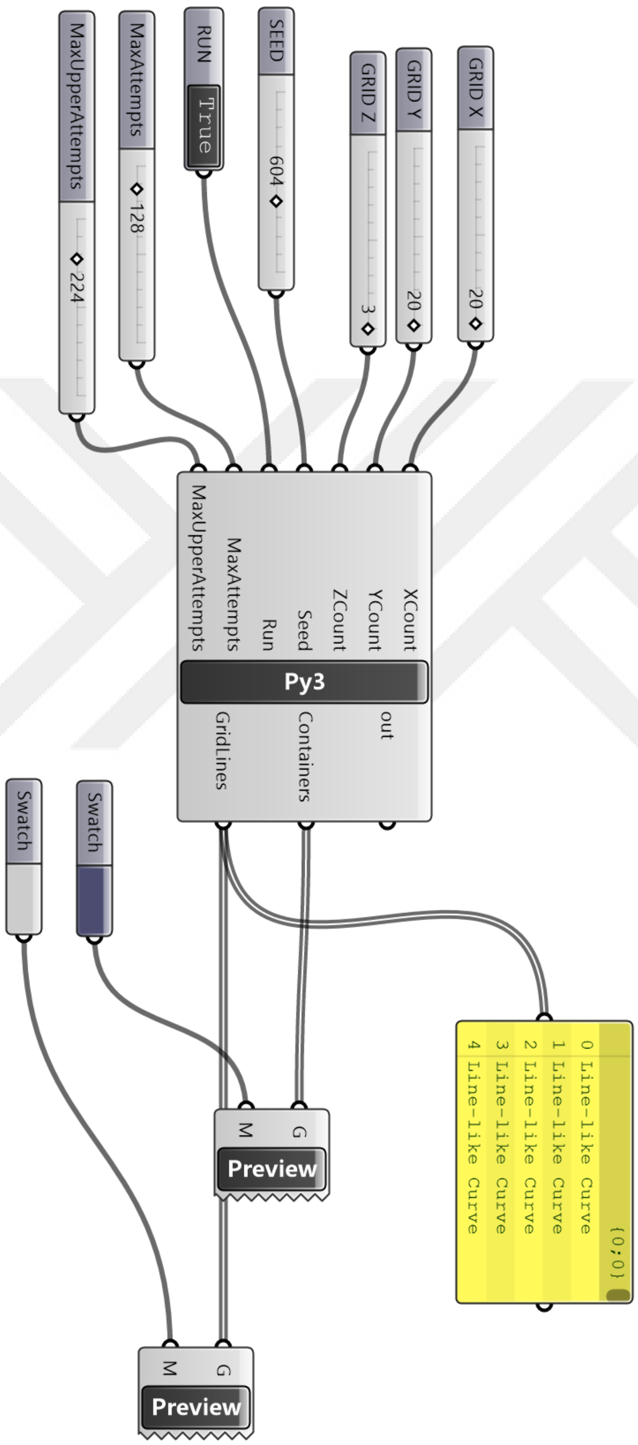


Figure A.1: Grasshopper elements.

APPENDIX B: Python Script

Appendix B contains part of the Python script used within the Grasshopper Python component to generate the container settlement configurations and CA rules described in Section 4.3, titled Rules of Settlement.

```
from collections import deque

def create_grid(x_count, y_count, z_count):

Produces a 3D cellular lattice.
Cell state: 0 = empty, 1 = filled

    return [[[0 for _ in range(z_count)] for _ in range(y_count)] for _ in
range(x_count)]

def neighbors_3d(x, y, z, x_count, y_count, z_count):

neighborhood (Von Neumann) definition: +X, -X, +Y, -Y, +Z, -Z
Local interaction rules in CA are written over this neighborhood.

    dirs = [(1,0,0), (-1,0,0), (0,1,0), (0,-1,0), (0,0,1), (0,0,-1)]
    result = []
    for dx, dy, dz in dirs:
        nx, ny, nz = x+dx, y+dy, z+dz
        if 0 <= nx < x_count and 0 <= ny < y_count and 0 <= nz < z_count:
            result.append((nx, ny, nz))
    return result

def connected_components(grid, z_level):

Finds components connected by 4-neighborhood (in the plan plane) on a specific
floor (z_level).
Cluster sizes are used in the concentration/clustering analysis in the CA output.

    x_count = len(grid)
    y_count = len(grid[0])
    visited = [[False]*y_count for _ in range(x_count)]
    components = []

    for x in range(x_count):
        for y in range(y_count):
            if grid[x][y][z_level] == 1 and not visited[x][y]:
                q = deque([(x, y)])
                visited[x][y] = True
                comp = []

                while q:
                    cx, cy = q.popleft()
```

```
                comp.append((cx, cy))
                for dx, dy in [(1,0),(-1,0),(0,1),(0,-1)]:
                    nx, ny = cx+dx, cy+dy
                    if 0 <= nx < x_count and 0 <= ny < y_count:
                        if grid[nx][ny][z_level] == 1 and not visited[nx][ny]:
                            visited[nx][ny] = True
                            q.append((nx, ny))
            components.append(comp)

    return components

def max_connected_size(grid, z_level):
```

Returns the number of cells in the largest connected component on a given layer.

```
    comps = connected_components(grid, z_level)
    return max((len(c) for c in comps), default=0)

def support_check(grid, x, y, z, x_count, y_count, allow_cantilever=True):
```

Vertical support rule (CA's z-axis constraint):
- If z == 0, no support is required.
- If the bottom cell is filled, support is required.
- If cantilevering is allowed (allow_cantilever=True), support is also considered required if any of the side neighbors at the bottom level are filled.

```
    if z == 0:
        return True
    if grid[x][y][z-1] == 1:
        return True
    if not allow_cantilever:
        return False
    for dx, dy in [(1,0),(-1,0),(0,1),(0,-1)]:
        nx, ny = x+dx, y+dy
        if 0 <= nx < x_count and 0 <= ny < y_count:
            if grid[nx][ny][z-1] == 1:
                return True
    return False
```

APPENDIX C: Java Script

Appendix C contains part of the Java script used within the Processing component. The complete code also includes the elements required for creating the GUI. In this section, only the CA rules originally written in Python and later translated into Java are shared.

```
public class ContainerCA {
   int X_COUNT = 20;   /
   int Z_COUNT = 14;
   int Y_COUNT = 3;

   long SEED = 42;
   int MaxAttempts = 1200;
   int MaxUpperAttempts = 1600;

   int[][][] grid;
   Random rng;

   public ContainerCA(int x, int z, int y, long seed) {
      this.X_COUNT = x;
      this.Z_COUNT = z;
      this.Y_COUNT = y;
      this.SEED = seed;
      this.grid = new int[X_COUNT][Y_COUNT][Z_COUNT];
      this.rng = new Random(SEED);
   }
```

Container settlement process

```
   public void generate() {
      grid = new int[X_COUNT][Y_COUNT][Z_COUNT];
      rng = new Random(SEED);
```

Ground floor (y=0)

```
      int attempts = 0;
      while (attempts < MaxAttempts) {
         int x = rng.nextInt(Math.max(1, X_COUNT - 1));
         int z = rng.nextInt(Z_COUNT);
         int y = 0;
         String dir = rng.nextBoolean() ? "x" : "z";
         int length = 2;

         if (!canPlace(x, y, z, dir, length)) { attempts++; continue; }
         placeContainer(x, y, z, dir, length);
         if (maxConnectedSize(y) > 8) removeContainer(x, y, z, dir, length);
         attempts++;
      }
```

Upper floors (y>=1)

```
        int upperAttempts = 0;
        while (upperAttempts < MaxUpperAttempts) {
            int x = rng.nextInt(Math.max(1, X_COUNT - 1));
            int z = rng.nextInt(Z_COUNT);
            int y = rng.nextInt(Math.max(1, Y_COUNT)); // 0..Y-1
            if (y == 0) { upperAttempts++; continue; }
            String dir = rng.nextBoolean() ? "x" : "z";
            int length = 2;

            if (!canPlace(x, y, z, dir, length)) { upperAttempts++; continue; }

            boolean supported = true;
            for (int i = 0; i < length; i++) {
                int xi = dir.equals("x") ? x + i : x;
                int zi = dir.equals("z") ? z + i : z;
                if (!supportCheck(xi, y, zi)) { supported = false; break; }
            }
            if (!supported) { upperAttempts++; continue; }

            placeContainer(x, y, z, dir, length);
            if (maxConnectedSize(y) > 8) removeContainer(x, y, z, dir, length);
            upperAttempts++;
        }
    }
```

Can a container be placed at this location?

```
    boolean canPlace(int x, int y, int z, String dir, int length) {
        for (int i = 0; i < length; i++) {
            int xi = dir.equals("x") ? x + i : x;
            int zi = dir.equals("z") ? z + i : z;
            if (xi >= X_COUNT || zi >= Z_COUNT || y >= Y_COUNT) return false;
            if (grid[xi][y][zi] == 1) return false;
        }
        return true;
    }
```

Place container

```
    void placeContainer(int x, int y, int z, String dir, int length) {
        for (int i = 0; i < length; i++) {
            int xi = dir.equals("x") ? x + i : x;
            int zi = dir.equals("z") ? z + i : z;
            grid[xi][y][zi] = 1;
        }
    }
```

Remove container

```
    void removeContainer(int x, int y, int z, String dir, int length) {
        for (int i = 0; i < length; i++) {
            int xi = dir.equals("x") ? x + i : x;
```

```
            int zi = dir.equals("z") ? z + i : z;
            grid[xi][y][zi] = 0;
        }
    }
```

Support check for upper floor containers

```
    boolean supportCheck(int x, int y, int z) {
        if (y == 0) return true;
        if (grid[x][y-1][z] == 1) return true;
        int[][] dirs = {{1,0},{-1,0},{0,1},{0,-1}};
        for (int[] d : dirs) {
            int nx = x + d[0], nz = z + d[1];
            if (nx >= 0 && nx < X_COUNT && nz >= 0 && nz < Z_COUNT) {
                if (grid[nx][y-1][nz] == 1) return true;
            }
        }
        return false;
    }
```

Maximum number of connected cells per floor

```
    int maxConnectedSize(int y) {
        boolean[][] visited = new boolean[X_COUNT][Z_COUNT];
        int maxSize = 0;
        for (int x = 0; x < X_COUNT; x++) {
            for (int z = 0; z < Z_COUNT; z++) {
                if (grid[x][y][z] == 1 && !visited[x][z]) {
                    int size = bfsCount(x, y, z, visited);
                    if (size > maxSize) maxSize = size;
                }
            }
        }
        return maxSize;
    }
```

Counting connected cells using BFS

```
    int bfsCount(int sx, int sy, int sz, boolean[][] visited) {
        ArrayDeque<int[]> q = new ArrayDeque<int[]>();
        q.add(new int[]{sx, sz});
        visited[sx][sz] = true;
        int count = 0;
        int[][] dirs = {{1,0},{-1,0},{0,1},{0,-1}};
        while (!q.isEmpty()) {
            int[] cur = q.remove();
            count++;
            for (int[] d : dirs) {
                int nx = cur[0] + d[0], nz = cur[1] + d[1];
                if (nx >= 0 && nx < X_COUNT && nz >= 0 && nz < Z_COUNT) {
                    if (!visited[nx][nz] && grid[nx][sy][nz] == 1) {
                        visited[nx][nz] = true;
                        q.add(new int[]{nx, nz});
```

```
                }
            }
        }
    }
    return count;
}
```

Test main

```
public static void main(String[] args) {
    ContainerCA ca = new ContainerCA(20, 14, 3, 42);
    ca.generate();
    System.out.println("Yerleşim tamamlandı.");
}
}
```

APPENDIX D: Renders.

Figure A.2: Relationship between public open spaces and containers.

Figure A.3: Neighbouring modules.

Figure A.4: View from a terrace.

Figure A.5: Fabric sunshades between modules.

CURRICULUM VITAE

Name Surname : **Hatice Seray ÖZBAY**

EDUCATION :

- **B.Arch.** : 2022, Middle East Technical University, Faculty of Architecture, Architecture

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2021 (2 months) A Tasarım Mimarlık, Ankara, Intern
- 2022 (2 months) Pasifik GYO, Ankara, Intern
- 2022 (continuing) Turkish Airlines, Istanbul, Specialist

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Özbay, H.S., Alaçam, S.** (2024). Examining Collaborative Design: A Critical Review Of Participatory Architectural Practices. *The 12th International Conference of the Arab Society for Computation in Architecture*, 13-15 November, 2024 Barcelona, Spain.