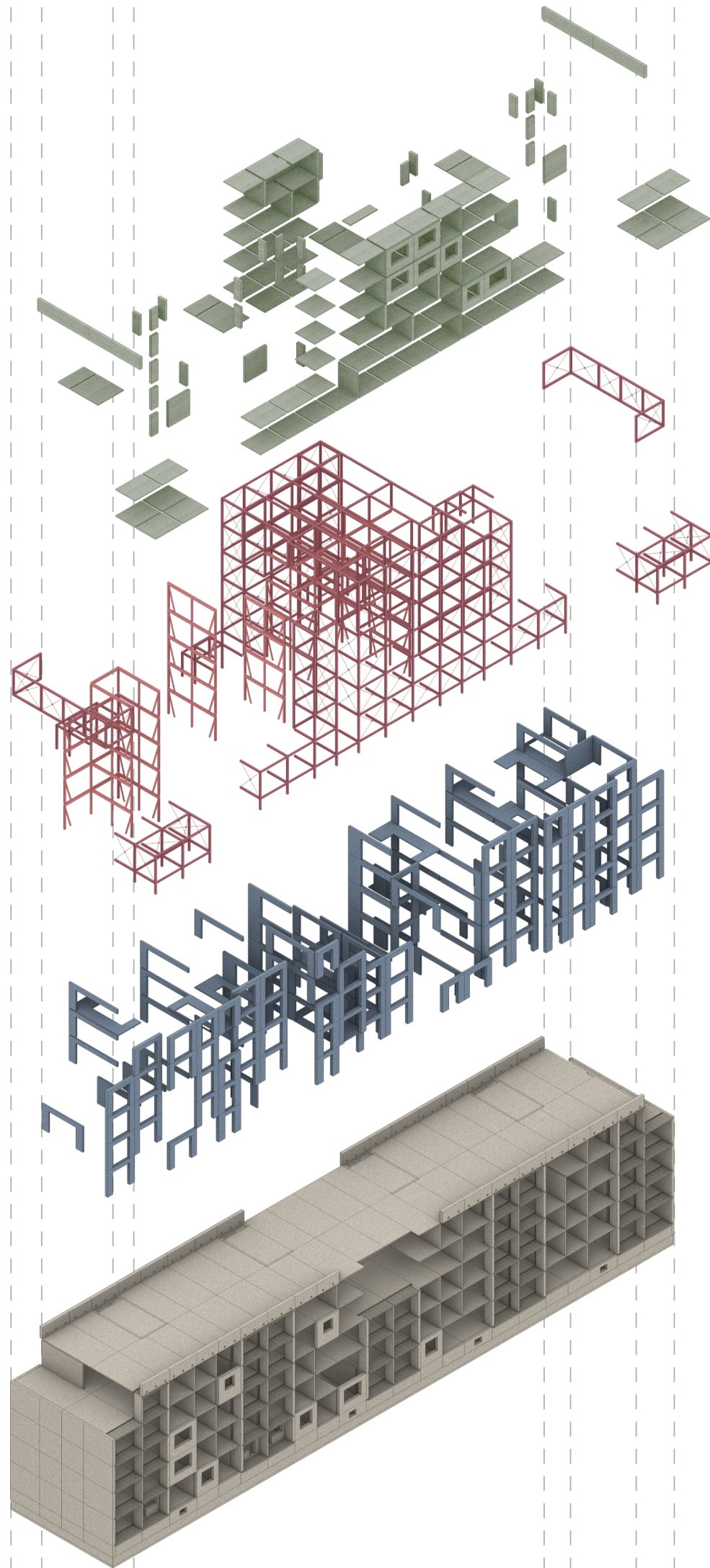




SYSTEMIC ADAPTATION OF SOVIET MASS HOUSING: A COMPUTATIONAL FRAMEWORK FOR SPATIAL AND STRUCTURAL REORGANISATION

Telma Terêze Lediņa

Degree Project in Digital Architecture and Emergent Futures
Lund University || Faculty of Engineering

Course: ASEM01

Supervisor: Gediminas Kirdeikis

Examiner: David Andréen

Spring 2026



LUND
UNIVERSITY

LTH

FACULTY OF
ENGINEERING

TABLE OF CONTENTS

1. ABSTRACT	2
2. INTRODUCTION	3
2.1 Thesis Roadmap and Objectives	5
2.2 Background	7
2.2.1 The Urban Context	7
2.2.2 The Expiration Date	7
2.2.3 The Socio-Economic Context	8
2.2.4 Site Focus and Urban Analysis	13
3. THEORETICAL FRAMEWORK	23
3.1 The Fallacy of Erasure	25
3.2 Architecture of Decay	27
3.3 The Circular Shift	29
3.4 Buildings as Material Banks	31
4. TYPOLOGICAL DIAGNOSIS: 1-464A SERIES	33
4.1 Technical Anatomy	35
4.2 Structural Vulnerabilities	37
4.3 Spatial and Architectural Anatomy	38
5. THE COMPUTATIONAL METHODOLOGY	43
5.1 Digital Cataloguing	45
5.2 Element Dependency Definition	49
5.3 Top-Down Versus Bottom-Up Logic	55
6. SYSTEMIC ADAPTATION	61
6.1 The Dependency Web	63
6.2 Element-Level Transformation Scenarios	65
6.3 Assembly-Level Transformation Matrix	69
6.4 Locked Versus Variable Rulesets	75
6.5 Grayscale Sampler as a Tool	77
6.6 The Combined Tool	79
6.7 User-Guided Workflow and Tool Integration	82
7. ITERATIVE TESTING AND SCENARIO EVALUATION	83
7.1 Experiment 1	85
7.2 Experiment 2	91
7.3 Evaluating the Spatial Consequences of Each Scenario	93
7.4 Experiment 3	101
7.5 Experiment 4	109
7.6 Experiment 5	115
8. ARCHITECTURAL SYNTHESIS	125
8.1 Translating the Algorithmic Output into an Architectural Outcome	127
8.2 Tectonics and Materiality	129
8.3 Structural, Architectural and Urban Consequences	129
9. CONCLUSIONS AND REFLECTIONS	161
9.1 Conclusions	161
9.2 Discussion and Reflections	163
10. ACKNOWLEDGEMENTS	165
11. TABLE OF FIGURES	165
12. BIBLIOGRAPHY	169

1. ABSTRACT

This thesis aims to address a dual crisis. First, a global crisis: there is a progressive ecological and sustainability concern regarding the current linear-model construction industry, which predominantly utilises a “take-make-dispose” approach, relying exclusively on the use of raw materials and a “tabula rasa” tactic in each novel project. This approach is bound to drain the Earth’s resources in the coming years and therefore needs to be reevaluated. Second, a local crisis: around a half of residential buildings in Riga, Latvia, are of Soviet mass housing typology. These structures are on the verge of structural expiration, at risk of progressive collapse, and are spatially obsolete and unfit for modern living standards. Consequently, typical methods of renovation are not enough to prolong their life expectancy; therefore, an innovative approach must be designed.

Conventional architectural methods, such as façade thermal insulation and minor interior repairs fail to address core spatial and structural issues due to the inherent structural complexity and rigidity of prefabricated concrete panel structures. Therefore, an extensive analysis of a case study building (a Soviet prefabricated concrete residential building of Series 1-464A) was conducted to evaluate its architectural possibilities. This evaluation subsequently enabled the proposal of a computational methodology for adaptive transformation, “cannibalisation,” and circular material reuse.

Since prefabricated concrete panel buildings are modular, computational methods are highly applicable. The first step was to isolate a specific object building of the 1-464A series and create a high-reliability digital twin of each modular element, as well as the overall assembly. Next, the structural scheme of the building was evaluated and relationships between modular elements were established, forming a structural dependency matrix. From this matrix, a custom semantic structural dependency script was developed in Grasshopper environment to enable precise structural reliance detection. Next, element-level scenarios were defined and grouped into four assembly-level scenarios, ranging from least to most structurally intrusive and cost-intensive. A

grayscale image sampler was integrated as an input tool to guide the application of these scenarios. The application of the scenarios uses the established dependencies to computationally determine forced alterations versus “free to alter” ones. Additionally, this system accounts for all material and elements removed from the building during the process to enable their circular reapplication in the architectural synthesis stage.

Through this tool, architectural interventions were tested in a series of experiments. In this process, the bridge between the architect as the user and the computational system was evaluated, and the system was refined to improve the user-guided experience. These experiments culminated in one final architectural synthesis output, proving that the system successfully enables spatial reorganisation - such as merging outdated units into modern, functional layouts or creating open floor plans for public functions - while retaining the structural integrity of the building. The methodology established within this thesis challenges the construction industry’s reliance on raw material consumption, simultaneously extending the life span of “expired” buildings.

2. INTRODUCTION

Ever since starting my architecture studies, while walking the corridors of the university, I have been seeing countless competition entries, theses, and research projects on adaptive reuse. This makes sense, of course, since I come from Latvia, specifically Riga – a city where the population has reduced critically over the last 35 years following the fall of the USSR. Many buildings, especially industrial ones, have been abandoned and stand as empty ghosts in the face of the cities. And architects often tend to see these large empty spaces as a pattern-breaking substance in the fabric of the city, therefore offering an insight on how to transform and adapt these spaces to the cities we envision.

However, in contrast to the countless theoretical proposals on this topic, only a few have manifested into real acts of adaptive reuse. For example, there have been more than 500 architectural idea competitions since 1991 in Latvia (97% or 130 of them in Riga) focused on residential typology. Yet, less than one-fifth of them have manifested into real buildings, or, at least, completed projects (Groskaufmanis & Ozola, 2019, p. 122). The reasoning behind this is one of the topics explored in this thesis, but my presumption is that it has something to do with the huge gap between what we as architects see as potential and necessary and what the rest of the society actually wants and needs.

This disproportion provokes a question: what about the buildings that we still live in, but that are bound to “expire” in the coming decade or two (or have expired already)? Is it possible to adapt these for modern living conditions? Or is it more sensible to tear them down and build anew? How about doing both - disassembling the existing structures and using the elements to build something semi-new, something that fits the current standard of living? Or something for a new function and typology entirely? It should be cheaper and more sustainable than producing everything from scratch. However, this approach is met with another obstacle - legislative norms and Eurocode standards for building materials. How can we be sure that something used more than half a century ago is still safe to use now?

Furthermore, due to the nature of construction processes, reusing materials from an existing building requires the creation of material banks. In this process, there is a window for the implementation of several types of digital tools and algorithms. This means exploring a type of workflow that allows for material banking and cataloguing before any physical deconstruction takes place, as well as exploring whether this is computationally possible at all.

This thesis therefore aims to investigate the typology of these “expiring” buildings and their adaptability potential. The buildings chosen for this project are the immediate candidates for becoming uninhabitable in the near future – Soviet prefabricated residential buildings. Because they are at risk of becoming unsafe to inhabit if no repairs or renovations are done on them, they make for the perfect research object for a theoretical experiment – if it is unsuccessful, the figurative “price of the loss” is low, as these buildings may soon collapse anyway.



Figure 1. Comparison between the original building facade and a facade altered using the developed computational framework

2.1 THESIS ROADMAP AND OBJECTIVES

Research aim

This thesis aims to address two critical, intersecting issues: the technical expiration and structural deterioration of Soviet era mass housing and the unsustainable volume of waste generated by the construction sector due to its reliance on the traditional “take-make-dispose” linear economic model.

I intend to offer a viable alternative to the current binary of demolition or simplified renovation. Since surface-level renovations often fail to address the obsolete spatial organisations and cramped conditions of these buildings, this research focuses on adaptive transformation.

The research specifically targets prefabricated concrete assemblies, as these are at the highest risk of structural instability and are more alterable through computational methods than other Soviet building typologies.

Research situation

The research is situated in Riga, Latvia, a city where I come from and have lived in all my life. Riga was chosen as the research focus because it is facing issues that I am seeing in my day-to-day life as both an inhabitant and an architect - shortage of quality housing, degrading buildings and stagnation of the existing renovation rate.

Riga represents an ideal “urban mine”, where buildings have technically exceeded their predicted lifespans but remain occupied. Focusing here allows for a nuanced analysis of specific local needs and socioeconomic aspects that must inform any 21st century architectural intervention.

The situation also provides a framework to investigate and revise legislative barriers that currently prohibit the reuse of structural elements and invasive spatial rearrangements.

Research object

The research object is a dominant, prefabricated concrete housing series in Riga (series 1-464A). The main criterion for the selection is both the availability of the original project documentation and the feasibility of conducting scans and visual documentation of the chosen series. By picking a dominant but high-risk series, the research maintains a realistic approach, focusing on a typology that is both “at risk” and computationally alterable.

Research question

By utilising computational algorithms, both the structural integrity/performance and the spatial organisation of a Soviet-period precast concrete residential housing typology could be altered and improved to ensure an extension of its lifespan, providing improvements of living space quality and preventing linear demolition

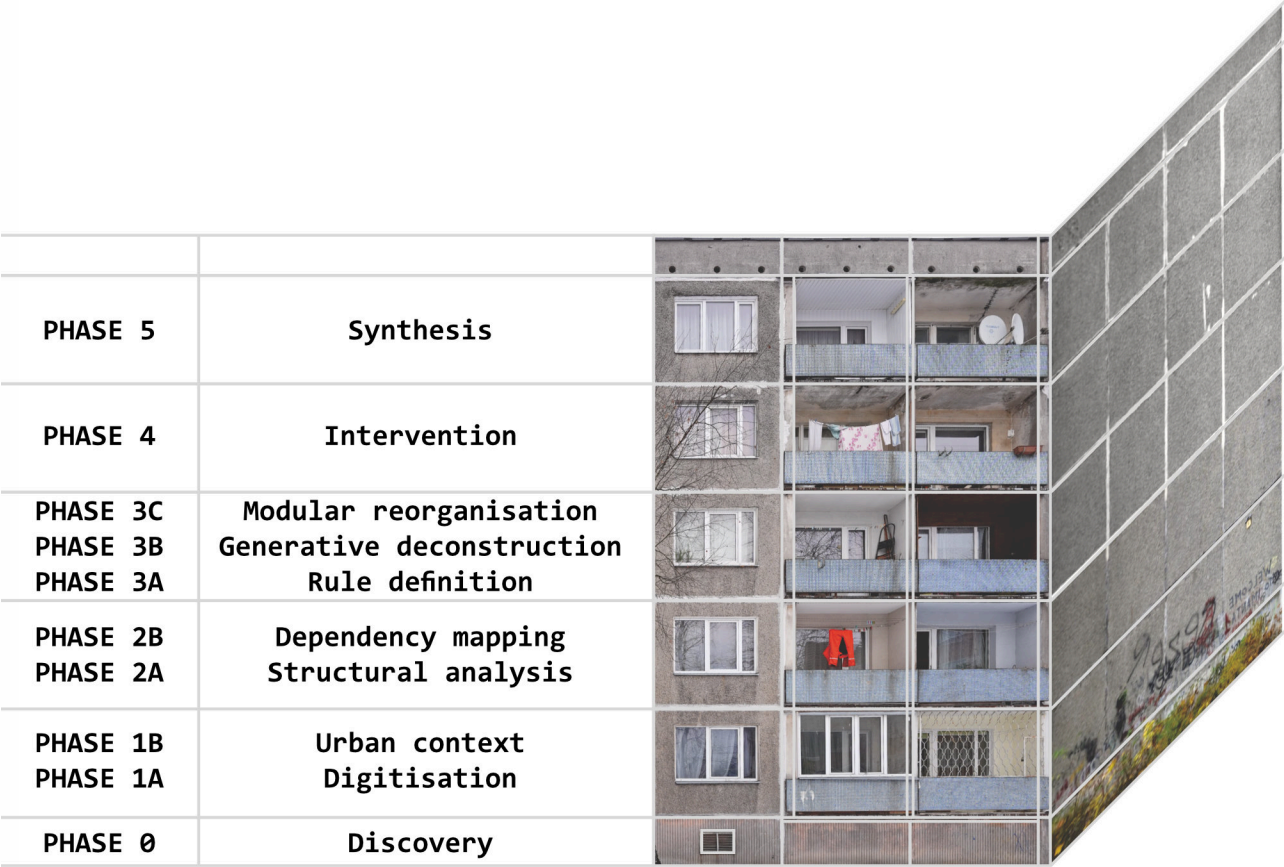


Figure 2. Proposed six-phase digital workflow outlining the thesis methodology

Research method

For this research, I propose a six-phase digital workflow designed to move from diagnosis to generative solution.

Phase 0 focuses on discovery and analysis – choosing a specific building series to work on, gathering archival and visual documentation, as well as selecting a preliminary site and area of Riga to focus the design output in.

Phase 1 consists of creating a high-reliability digital twin of the chosen prefabricated concrete building, element cataloguing, and analysing the assembly sequence of the original building, which is relevant for the disassembly approach. An urban analysis of the chosen neighbourhood to identify urban-scale local deficiencies is also included in this phase.

Phase 2 focuses on micro-scale relations within the existing building, inspecting existing technical inspection reports on the building series. From that, all the dependencies between the building elements are mapped to figure out the structural relations and dependencies between the building elements.

Phase 3 is where the intervention system stands – in this phase, the scenario definition is carried out, rule-sets are created and the script is written to tie that together with the existing dependencies.

Phase 4 is where the created tool is tested through various experiments and iterations.

The last phase – phase 5, is the architectural synthesis where the geometry, carved and reorganised through the tool, is synthesised and transformed back into a fully functional building with new spatial and architectural value.

2.2 BACKGROUND

This section establishes the background of the chosen research focus in Riga, Latvia. It briefly examines the residential housing condition and its history, while investigating the typological specifics of Soviet residential buildings and the socio-economic context of their construction. This section also includes an urban analysis of a selected neighbourhood in Riga - Ilģuciems - to ground the subsequent algorithmic alteration framework within a tangible, real-world context.

2.2.1 THE URBAN CONTEXT

Residential housing has been an issue in Riga ever since the industrialisation of the city in the mid-19th century. Large quantities of workforce started entering the city from the countryside to work in the newly established factories, as well as the port, and the rate of construction of new housing units simply could not keep up. Up until World War I the growth of both the manufacturing and therefore the inhabitants were steady and rapid. During the war, in 1915, more than 85% of the factories alongside around 200 thousand people - workers, including their families - were “evacuated” to Russia (Rubīns, 2004, p. 8). In 1918, after World War I, newly established Republic of Latvia faced all kinds of challenges, and the impact of war had significantly affected the number of residents in the capital. During this time, first free-state period as it is called, everything in the country had to be created from scratch, including legal framework, political system, economics, international relations, and, from more physical point of view, the country had to also be rebuilt after the damage of the war itself.

During this “first independence” period, neither the economic situation nor the number of residents never managed to revert to the pre-war situation. With the start of World War II in 1939 any ongoing projects or procedures were put to stop. Further on, Latvia was occupied by Soviet Union in 1940, then by Nazi Germany in 1941, and by Soviet Union again in 1944. The war had not even come to an end when the Soviet Union started organising widespread immigration into

Latvia, especially Riga, from everywhere across the USSR. This rapid increase of people therefore amplified the issue of housing availability that had never been fully solved. This is where the Soviet mass housing comes in play – initially planned to be the solution of housing crisis all through the cities and nations of USSR. Of course, this ambition was never fulfilled and at the time of the collapse of Soviet Union (1990), there were around 200 thousand people or around 77 thousand families in Riga alone “in the line” to get an apartment, and from the average calculations, it would take them around 20 years to do so (Groskaufmanis & Ozola, 2019, p. 73).

2.2.2 THE EXPIRATION DATE

Fast-forward 35 years, the housing issue is as significant as ever. The population of both Riga and Latvia keeps diminishing due to the ageing of society and economically driven emigration, but there is still a huge gap in affordable, good quality housing. Most of the people in Riga currently live in Soviet mass housing and it is either already expired or about to expire, as the average lifespan of this housing typology was predicted to be 60 – 70 years (Zariņa & Uzulēns, 2023, p. 258). Since around 65% of inhabitants of Riga live in this housing typology (nbhd, 2023). In a very pessimistic scenario that means that in the coming decade, around 400 thousand residents of Riga could lose the roof over their head because these buildings will start to collapse and become unsafe for living in (that has already happened in individual cases). The good thing is that in the Soviet Union everything in construction was calculated with quite a bit of a margin, therefore these buildings will probably stand and be safe to inhabit for another 50 years.

Unfortunately, “safe to inhabit” does not correspond with the modern standards of living, and Latvia is already very much behind on square footage per person in an apartment. In comparison, data from 1993 shows that the square footage per person in Sweden was on average 43.4m² compared to 20.7m² in Latvia (Rubīns, 2004, p. 33). That is almost two times less, and even though the data is 30 years old, not much has changed in Latvia as the housing stock remains almost the same. Another issue is the micro climate in this housing typology – the buildings back

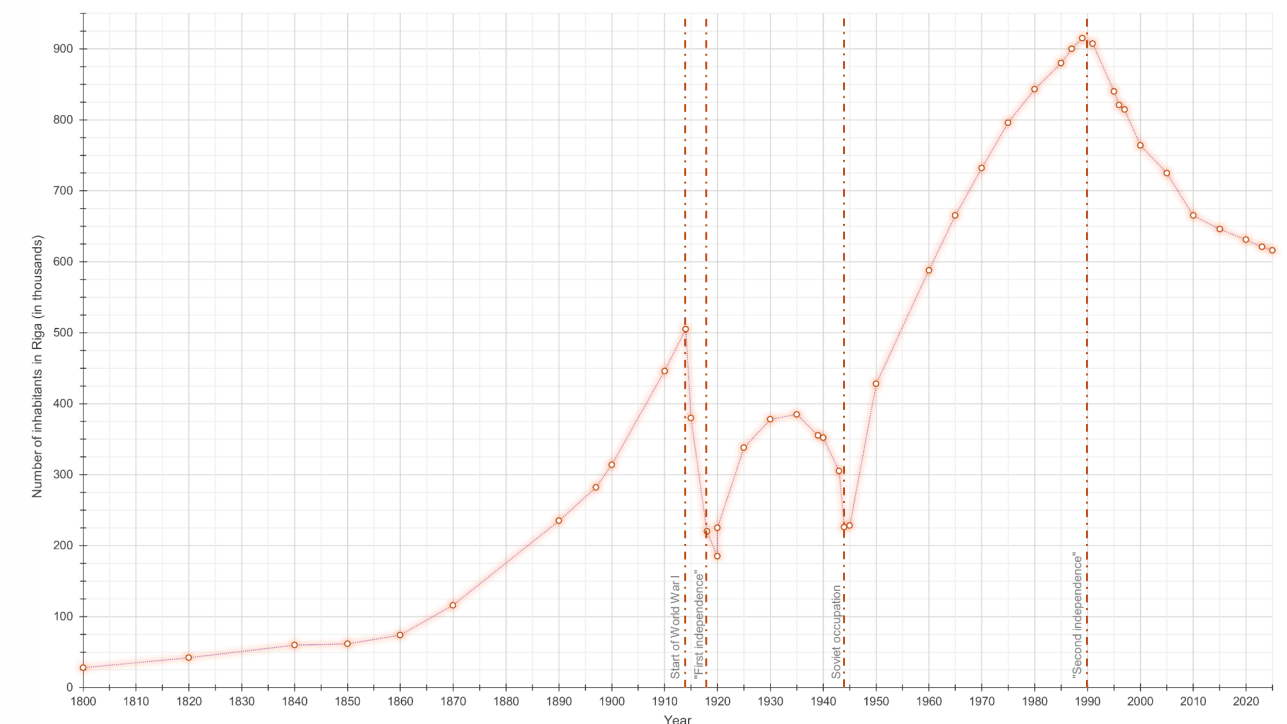


Figure 3. A chart showing population changes in Riga since the 19th century (Rubīns, 2004, p. 9)

then were built with no concern regarding energy efficiency or environmental impact, meaning there was no reason to thermally insulate the buildings as there were “unlimited” resources to heat them in winter. Currently this disregard towards energy conservation leads to either uncomfortable living conditions both in winter and summer and / or very high energy bills. The energy inefficiency of this housing typology goes against current energy conservation policies, and the living conditions in these buildings are obsolete, therefore rapid and radical change should be implemented.

2.2.3 THE SOCIO-ECONOMIC CONTEXT

Despite living in a Soviet concrete panel residential building only during my early childhood, I retain vivid memories of the daily realities within them. I distinctly recall the sound of an extremely loud TV that was always on. From an elderly neighbour across the hall, behind closed doors. There was also a pervasive smell of cigarette smoke in the

stairwell whenever the apartment door was opened, and the recurring issue of the sewage system backing up since our apartment was on the ground floor. Even today, whenever I visit my grandmother who lives in a similar building and sit down on her couch, I can clearly hear the conversations and the dog barking in the apartment next door through an electrical outlet in the wall. Anyone who has lived in this type of housing for some period of time could likely share similar or worse stories; however, the reality is nuanced. If the experience was always negative, people would have moved away long ago – yet they remain.

The method with which Soviet mass housing was developed and evolved into what can be seen today was very deliberate. The process was controlled by the state and any deviations from the central plan required approval from multiple bureaucratic levels. The whole mass housing development process in the Soviet Union, spanning fifty years, can be categorised into three distinct generations: the first from the mid-1940s to mid-1960s, the second from the mid-1960s to mid-1970s and the third from the mid-1970s

up until the collapse of the Soviet Union. While these generations followed a natural development pattern, they are distinguished primarily by the use of materials (shifting from brick in the first generation to large-scale prefabricated concrete elements later), the spatial organisation of the buildings in the city (moving from “random” placement to intentional *microrayons* or micro-neighbourhoods) and the general layout of the buildings and apartments (Treija & Bratuškins, 2019, pp. 171–173).

An analysis of these generations and the pattern of development reveals several core ideologies driving the Soviet mass housing construction process. Primarily, the goal was to provide each family with a unit, thereby improving overall housing conditions (Treija & Bratuškins, 2019, p. 163). Simultaneously, the state sought to optimise construction processes to the maximum extent possible, ensuring both low costs and fast assembly, through the rapid and extensive development of standardised prefabricated large panel systems (Meuser et al., 2015, p. 54), therefore ensuring the fulfilment of first ideological point faster and cheaper. Crucially, the third objective was maintaining social control by homogenising the living environment. The creation of standardised projects was not only a way of keeping down the costs, but also a way to ensure egalitarianism, enforcing the ideal that everyone was “equal” in the Soviet Union (with the notable exception of high-ranking political party members and other more “equal” citizens). This manifested in specific spatial strategies, for instance, kitchens were intentionally minimised to prevent large gatherings and the discussion of “secret” matters, thereby mitigating potential threats to the state (Groskaufmanis & Ozola, 2019, p. 12). Instead, people were encouraged to spend time outside of their private space in public venues. This ideology extended beyond architecture to domestic life – every apartment had the same furniture, same plates and cups, same cutlery and other everyday items. And, while regional differences existed across the vast Soviet territory, the underlying ideology remained constant.

Another social aspect are the above-mentioned *microrayons* - a “city within a city”, designed to minimise the need to ever leave one’s neighbourhood. All services and necessities – grocery stores, kindergartens,

schools, and doctors’ offices - were provided within these clusters (Treija & Bratuškins, 2019, p. 167). Combined with the green common spaces surrounding the residential building blocks, this created an “ideal” living space that essentially functioned as closed loop, again ensuring control over the population. And, while these *microrayons* appear comparable to community housing clusters in Western countries, they never ended up functioning as such, at least in Latvia. The failure likely stems from the nation’s earlier history and cultural mentality. Latvians are generally reserved and introverted, therefore, even though forced upon, the idea of communal living never took root here. The pinnacle of the life in Latvia during Soviet times was securing one’s “own” apartment, and that aspiration remains unchanged today. Although ownership has shifted from the state to the individual, the core desire persists: to secure a private space that belongs solely to the individual.

Following the restoration of independence, a comprehensive privatisation process of government-owned properties commenced, during which approximately 99% of the state-owned housing stock was offered for private acquisition. This has resulted in a highly complex system of fragmented ownership, where individual residents possess their specific units alongside a collective share of the building’s primary structure – including facades, roofs and communal spaces – and occasionally the underlying and / or surrounding land plot (Treija & Bratuškins, 2019, p. 174). This dispersed ownership structure complicates maintenance and renovation efforts, as the common property is frequently perceived as a collective burden rather than an individual priority. This “tragedy of the commons” has become a significant obstacle to upkeep and modernisation of Soviet-era buildings, effectively accelerating their technical expiration.

When evaluating these problematic social and structural factors, it might seem logical that this housing typology would be undesirable; however, certain series and *microrayons* remain among the most sought-after residential areas in Riga even today. This endurance is the result of several practical factors. First, the persistent housing shortage in the city means that a mass departure from the Soviet housing stock would inevitably lead to widespread housing

	First generation	Second generation	Third generation
Example series	 Series 118 building in Āgenskalna Priedes, Riga. Source: Uģis Bratuškins	 Series 467A building in Dzirciems, Riga. Source: author	 Series 602 building in Ziepniekkalns, Riga. Source: Laurijs Svirskis
Example spatial layout	 Spatial organisation of Āgenskalna Priedes, Riga	 Spatial organisation of Dzirciems, Riga	 Spatial organisation of Ziepniekkalns <i>microrayon</i>, Riga
Key characteristics	• Clear and simple multi-storey designs; • Silicate brick construction, also first prefabricated concrete panels in later years; • Small living rooms and minimum auxiliary space; • No outdoor spaces or very small balconies; • Smaller scale buildings (2 - 5 stories); • Simple, “free” spatial layouts with little attention to vertical composition or interaction between buildings. (Treija & Bratuškins, 2019, p. 171)	• Residential buildings in Riga get modernised and the projects are adapted to local conditions; • Most buildings are constructed from large-slab prefabricated concrete panels; • The apartments become more spacious, with more exterior space in form of balconies, more auxiliary spaces are added; • Larger scale buildings (5 - 9 stories); • More complex spatial arrangements that focus “inwards” towards green spaces, first creation of <i>microrayons</i>. (Treija & Bratuškins, 2019, p. 172)	• Building series are mostly designed by local architects and adapted to local needs; • Most buildings are constructed from large-slab prefabricated concrete panels; • The apartments are around 5% more spacious than the ones in 2nd generation; • The buildings are mostly large scale (9 - 12 stories); • The <i>microrayon</i> spatial arrangement is fully defined, creating ordered spatial organisations with regards to vertical composition and relationship to the city skyline. (Treija & Bratuškins, 2019, p. 172)

Table 1. Comparison of different residential building construction periods in Riga and their key spatial characteristics

instability. Furthermore, the significant price disparity between privatised older units and contemporary new builds creates a substantial barrier to mobility. Market data indicates that an apartment in a newly constructed residential building of similar scale and comfort often costs at least twice as much as an existing Soviet-era apartment (Valsts zemes dienests, 2025). The price differences between apartments in new and old buildings in the last two years are shown in Figure 4. Consequently, many residents face a choice between downsizing to a significantly smaller modern unit, assuming an unmanageable financial debt, or remaining in a familiar, albeit imperfect, environment. Most people understandably opt for the latter. Additionally, a typological preference often influences this decision. While modern developers frequently promote open-plan layouts, combining kitchens and living areas to maximise square footage efficiency, many residents continue to value the functional separation of a dedicated kitchen, regardless of its size. As a result, many people choose to stay in their current apartments, regardless of its impracticality or low quality, due to a combination of economic necessity and a preference for traditional spatial layouts that are more accustomed.

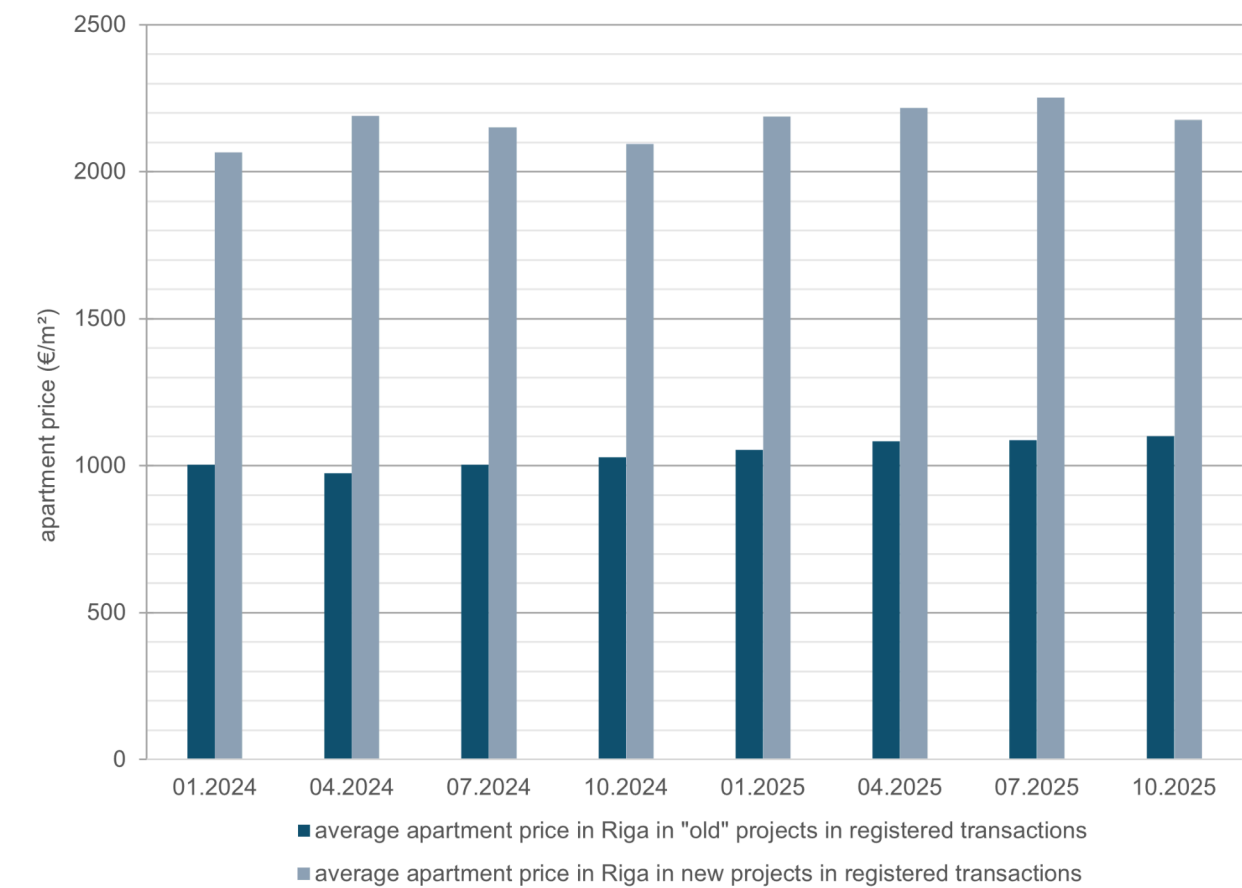


Figure 4. A comparison chart showing average “old” apartment prices compared to prices in new residential projects in the years 2024 and 2025 (Valsts zemes dienests, 2025)



Figure 5. Different ad-hoc loggia solutions in 1-464A series buildings in Ilūciems

2.2.4 SITE FOCUS AND URBAN ANALYSIS

For this research, a neighbourhood in Riga – Iļģuciems – was chosen as a case study due to several factors. Firstly, having lived near the area for most of my life provides an intimate understanding of the neighbourhood, making it possible to analyse it from the perspective of a local, not just as an architect. First-hand experience reveals that many people from this neighbourhood commute in and out of it daily for work, services and leisure. Secondly, the soviet housing typology is largely predominant in this neighbourhood, and even more so – the chosen series 1-464A constitutes a visibly large portion of that. Finally, the neighbourhood itself is large enough to theoretically be self-sufficient and include all the typical functions and categories of an urban area, and with this analysis that assumption is to be either proved or disproved.

The neighbourhood is located in the Western part of Riga, on the left bank of the river Daugava. It neighbours the Spilve, Dzirciems and Imanta neighbourhoods. Historically, it developed as a separate town apart from the main territory of Riga, including several free-standing manors, but during the industrial development in the 19th century it became a significant industrial area of the city periphery, including factories and residential units for the workers. It was included in the administrative boundaries of Riga municipality in 1924. The boundaries of the neighbourhood naturally have fluctuated throughout time, but currently the area of it is around 9,5 square kilometres and was inhabited by around 20,5 thousand inhabitants in 2024 (Sarnovičs, 2026).

The urban analysis demonstrates that the fundamental logic of a *microrayon* technically has been preserved in Iļģuciems, with basic functions of everyday life identifiable in the area. However, a more in-depth analysis of the micro-scale would be required to understand the daily routine of the neighbourhood completely, as non-residential buildings rarely possess a singular function. In this analysis, only the primary function of each building has been identified. To gain a deeper understanding, further research detailing specific commercial availability, service distribution, and workplace typologies would be required. Moreover, public buildings

and their functions often exceed the boundaries of a single neighbourhood, therefore the adjacent areas would typically be considered. However, within the scope of this project, it is assumed that the primary function of the building represents its sole program, and the neighbourhood is treated as an isolated system. If a function does not exist within the chosen boundaries, it is considered entirely absent for the local residents.

A closer examination of Iļģuciems reveals its current state as a mono-functional “sleeper district,” heavily dominated by residential typologies. Unlike in certain East Asian living cultures where apartments are often utilised solely for sleeping and social activities occur outside of these spaces, the harsh climate, as well as the socioeconomic aspects of the of the Baltic region, dictate that living spaces must also accommodate cooking, social gatherings, family time, work, sports and leisure activities. Therefore, these living spaces are [should be] not just for sleeping in, but for living in.

The proximity of the river Daugava, as well as the railway system, has consequently injected an industrial function in the neighbourhood. While this typology does provide local workplace opportunities, it simultaneously intensifies heavy traffic and contributes to urban degradation of the area. Additionally, the analysis reveals a significant lack of cultural and community infrastructure - a singular cultural centre, acting as an event hall, exhibition space, and library - is severely insufficient for roughly 20 thousand people inhabiting Iļģuciems.

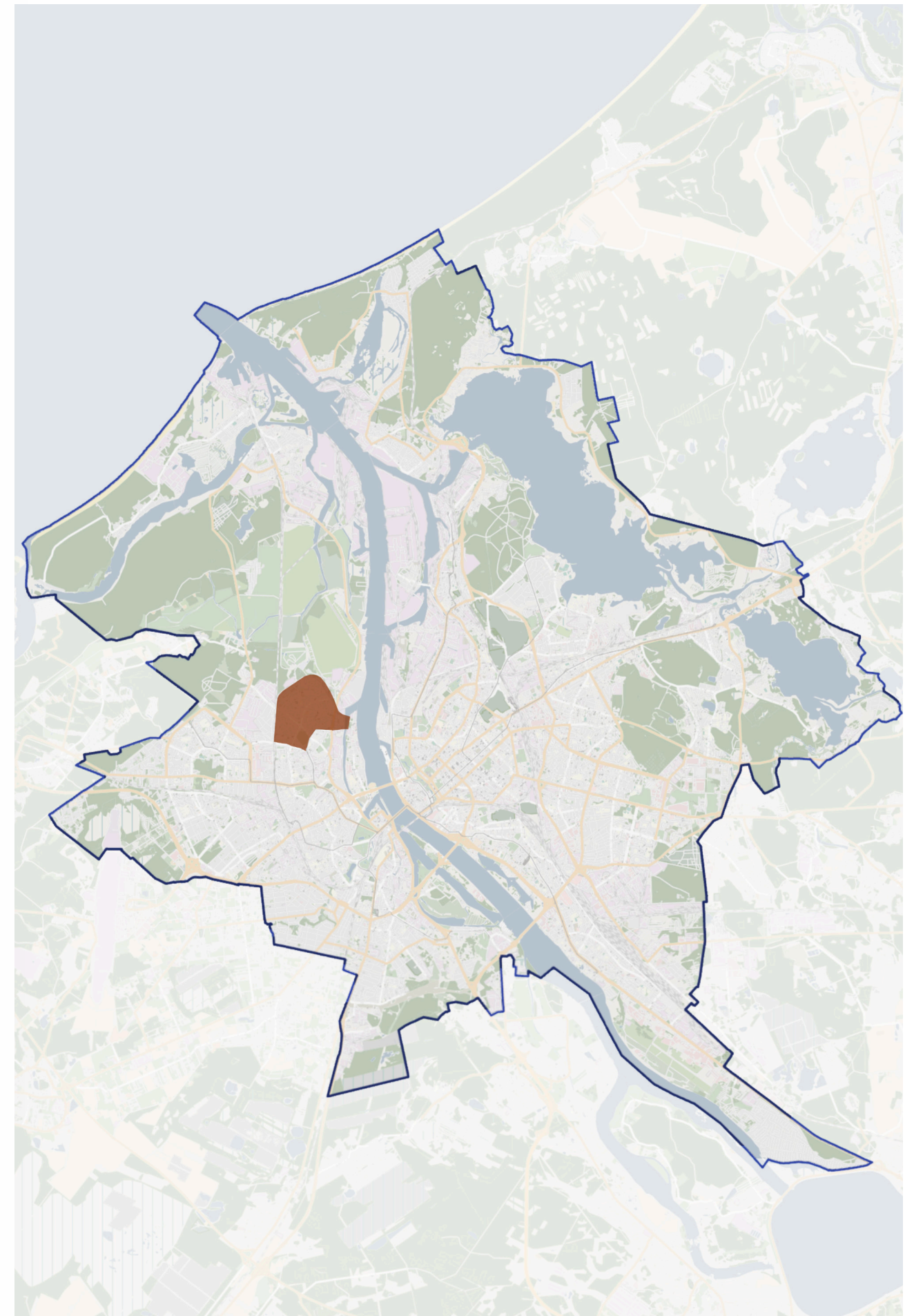


Figure 6. Location of Iļģuciems within Riga

Residential

Within the analysis, residential buildings are categorised into two major categories based on legislative definitions: detached single-family houses, and multi-family apartment buildings. To serve the scope of this research, apartment buildings are further sub-categorised into three groups chronologically: pre-1940, Soviet-era, and post-1990. Crucially, the Soviet-era buildings of series 1-464A, as they are the primary research object, have been isolated to highlight their morphological prevalence.

Visual inspection of the data confirms the neighbourhood's predominant residential character. Of the 275 apartment buildings in Ilūciems, 189 (over 65%) are Soviet built, while only 8 (under 3%) have been constructed in the past 35 years. Furthermore, 82 of these Soviet residential buildings (roughly 45% of the Soviet typology and around 30% of all apartment buildings) are of 1-464A series, emphasising their spatial dominance and the urgency of architectural intervention. This data empirically highlights the issue of outdated living conditions in the neighbourhood, firmly grounding the research relevance.



Figure 7. Different series of Soviet period precast concrete panel residential buildings in Ilūciems

Civic

This category includes healthcare facilities (one polyclinic across two buildings), cultural buildings (one cultural centre housing a library, exhibition hall, and spaces for community activities), government buildings (one municipal police station), and educational facilities (16 schools and kindergartens).

This typology marks the most significant functional deficit in the neighbourhood. While a single healthcare establishment may be sufficient for the area, as it provides a service that is not an everyday need, one cultural building is drastically insufficient for the entire district. There is a severe lack of spaces designed for community engagement, indoor workshops, and group activities. Consequently, these essential social functions are often forced into unsuitable environments, such as office spaces or cramped apartments. Furthermore, there is a noticeable deficit of indoor recreational facilities, such as gyms or yoga studios, which are currently limited to a few ad-hoc spaces integrated into commercial buildings.

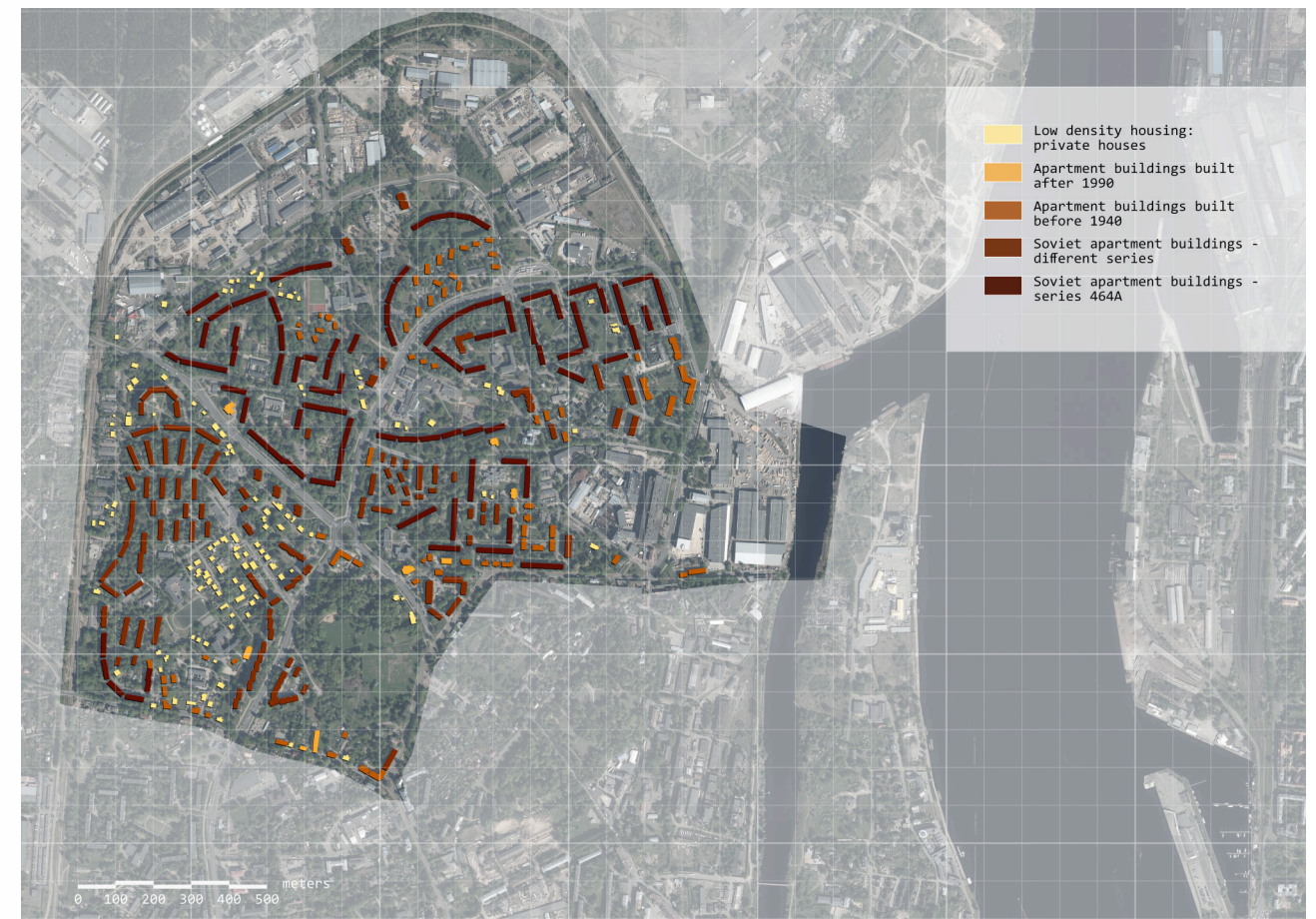


Figure 8. Residential building typology distribution in Ilūciems

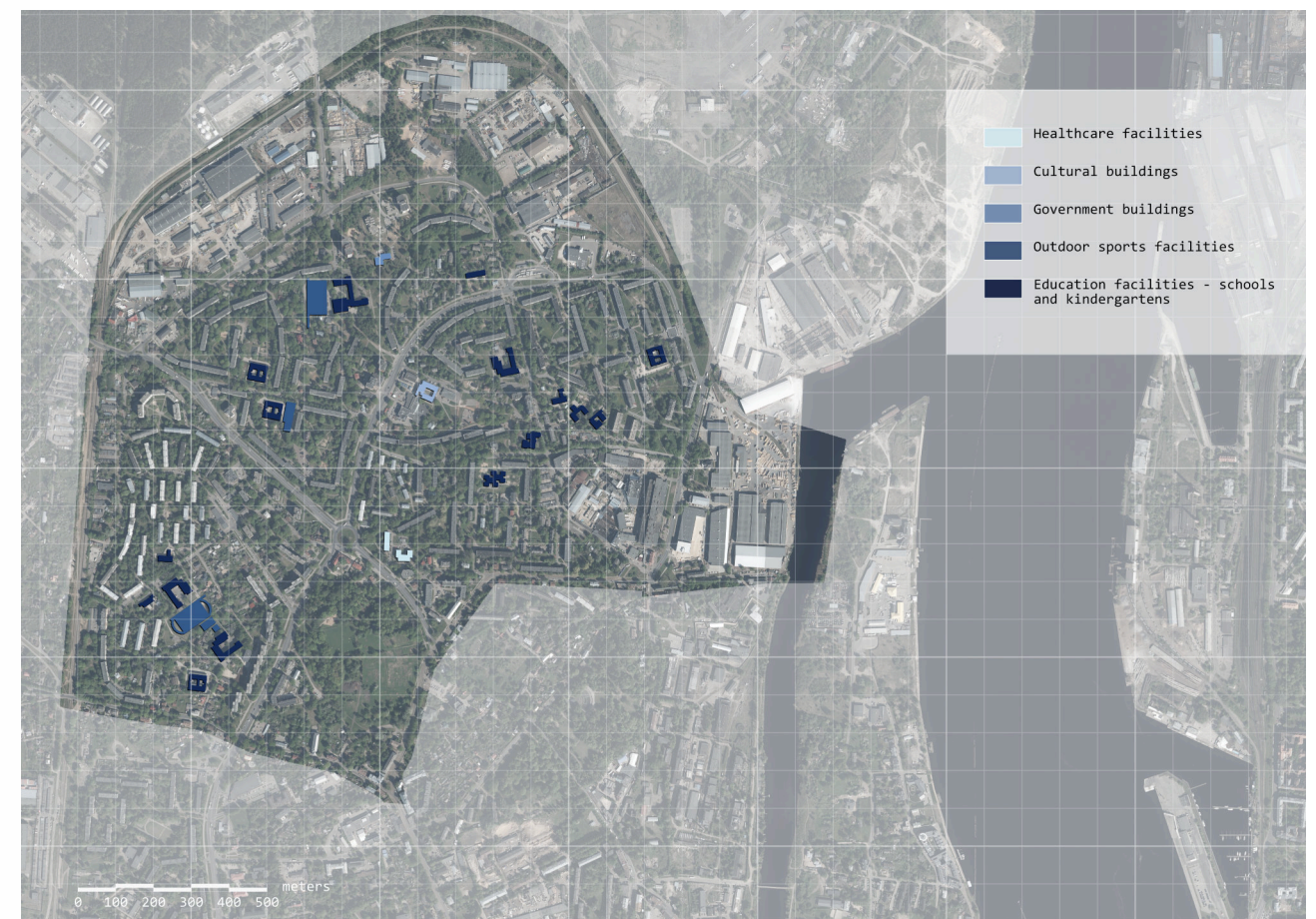


Figure 9. Civic building typology distribution in Ilūciems

Commercial

This category has been divided into two sections: dedicated office buildings (which may integrate minor services), and retail buildings (grocery stores, pharmacies, repair shops, beauty salons, etc.). While the retail services are distributed relatively evenly throughout the neighbourhood, the office buildings are predominantly clustered near industrial zones, indicating a symbiotic relationship with local manufacturing. This reveals a lack of diverse, modern workplace opportunities within the residential core. To prevent the daily migration of residents from Iļģuciems, new commercial typologies should be introduced. This could manifest as decentralised co-working spaces or dedicated remote-work zones integrated directly into the residential buildings.

Industrial

Proximity to both the port and the railway has systemically created two distinct industrial zones in the North and East of Iļģuciems, with the latter also having direct access to the waterway. While beneficial for local employment, these zones also introduce an industrial aesthetic that degrades the residential urban quality, and the associated transportation logistics create additional traffic congestion.

Conversely, the neighbourhood's garage cooperatives represent a highly unique architectural and urban typology, clustered into two primary zones within Iļģuciems. In Latvian and post-Soviet urban culture, these garages are rarely used for their intended function of vehicle storage, as they tend to be relatively far away from the respective apartments. Instead, they serve as decentralised community spaces for workshops, storage, and social gatherings. This local adaptation further underscores the critical lack of formal community spaces mentioned previously. A key architectural question emerges from this typology: could the social and cultural programs currently housed in these peripheral garages be integrated into the apartment buildings themselves?

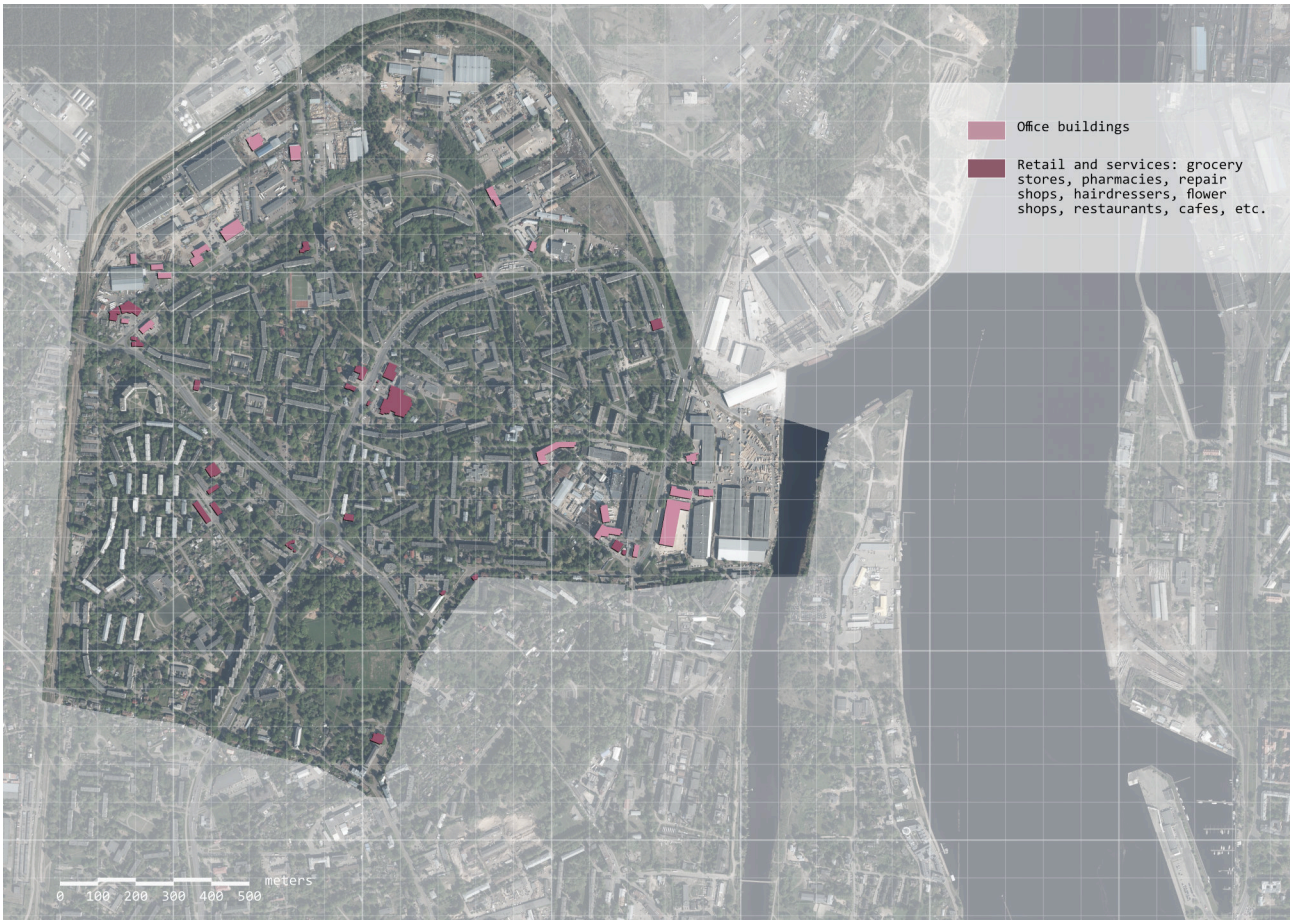


Figure 11. Commercial building typology distribution in Iļģuciems

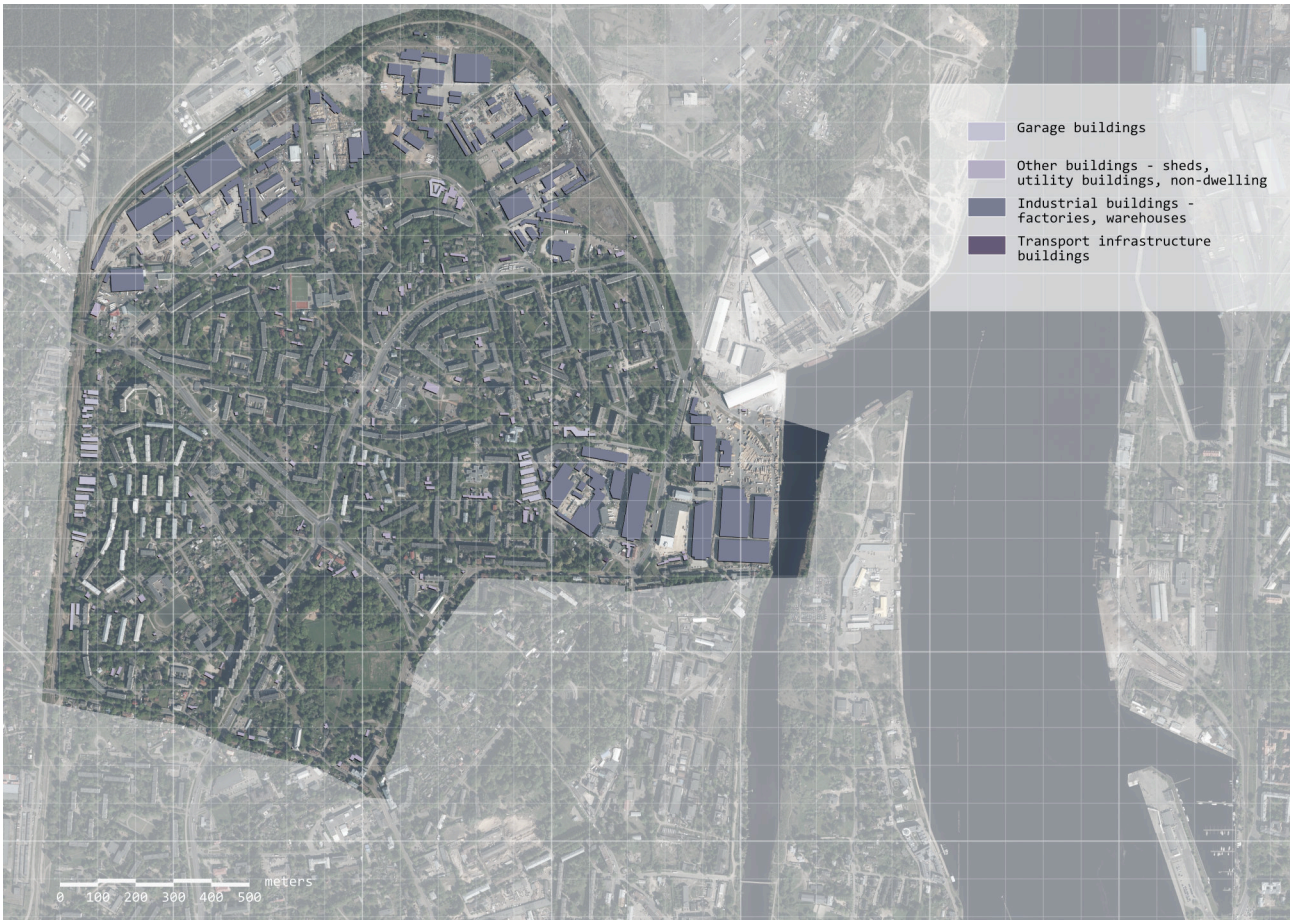


Figure 12. Industrial building typology distribution in Iļģuciems



Figure 10. Different series of Soviet period precast concrete panel residential buildings in Iļģuciems



Roads and mobility

The primary road network of Iļģuciems is dictated by its geography. While three distinct main roads connect the neighbourhood towards the city centre across the Daugava River, they ultimately converge at a single primary bridge crossing. This infrastructural bottleneck generates massive daily traffic jams, especially in the peak-hours. This issue stems from the local unavailability of amenities and services. Consequently, residents are forced to commute in and out of the neighbourhood to access these amenities, worsening the traffic congestion. That is exactly what was intended to be avoided with the creation of *microrayons*, but, after the collapse of Soviet Union, and the natural course of Riga's development, most of the functions have moved to the city centre, while the living spaces are still on the outskirts. To resolve this, everyday necessities and workplaces must be integrated into the neighbourhood, fulfilling the original, yet currently failing, conceptual intention of the *microrayon*.



Figure 13. An early experimental variation of precast concrete residential buildings in Iļģuciems

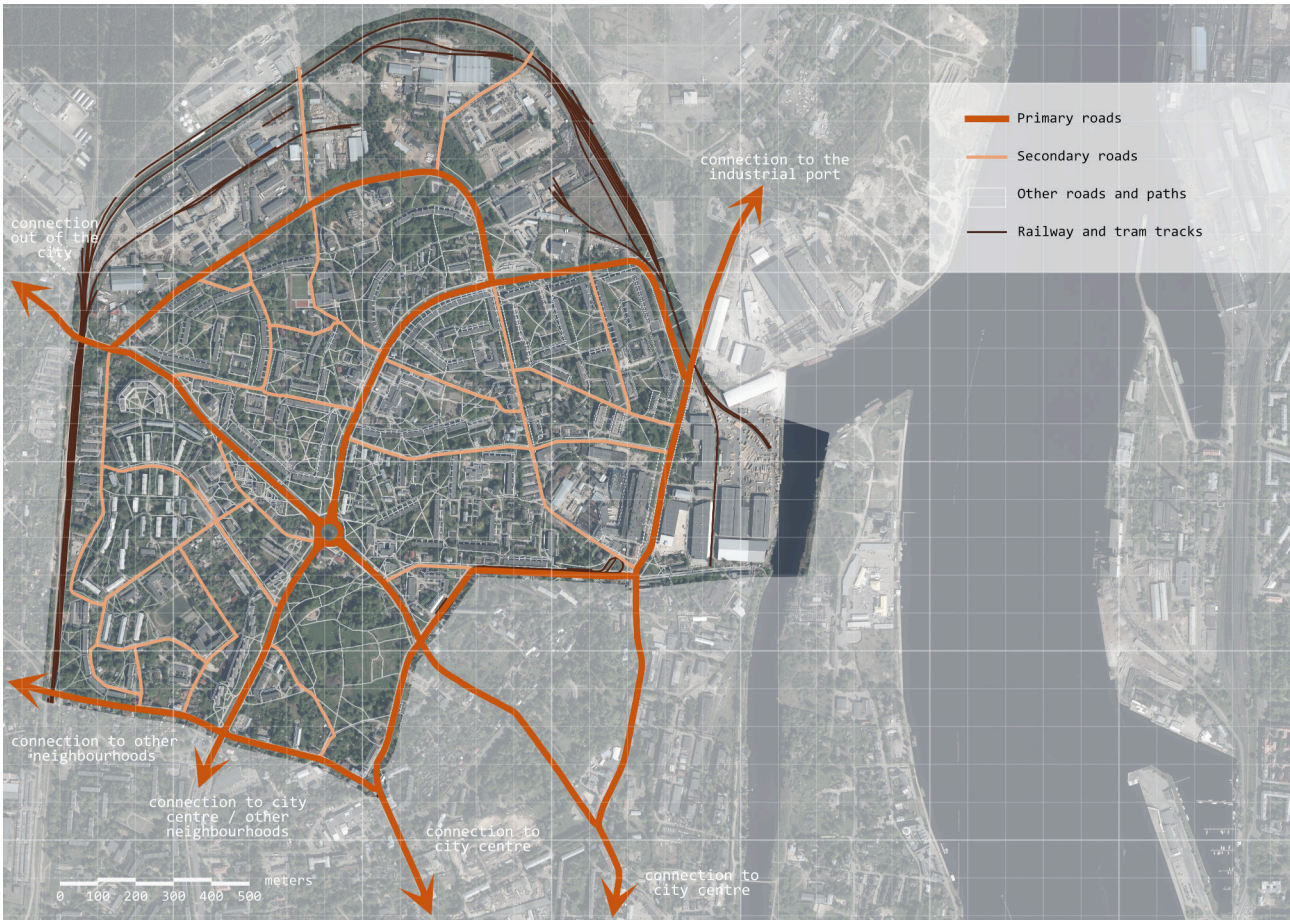


Figure 15. Road network in Iļģuciems



Figure 14. Different variations of the 1-464A series residential buildings in Iļģuciems



Figure 16. A variation of the 1-464A series residential building in Iļģuciems

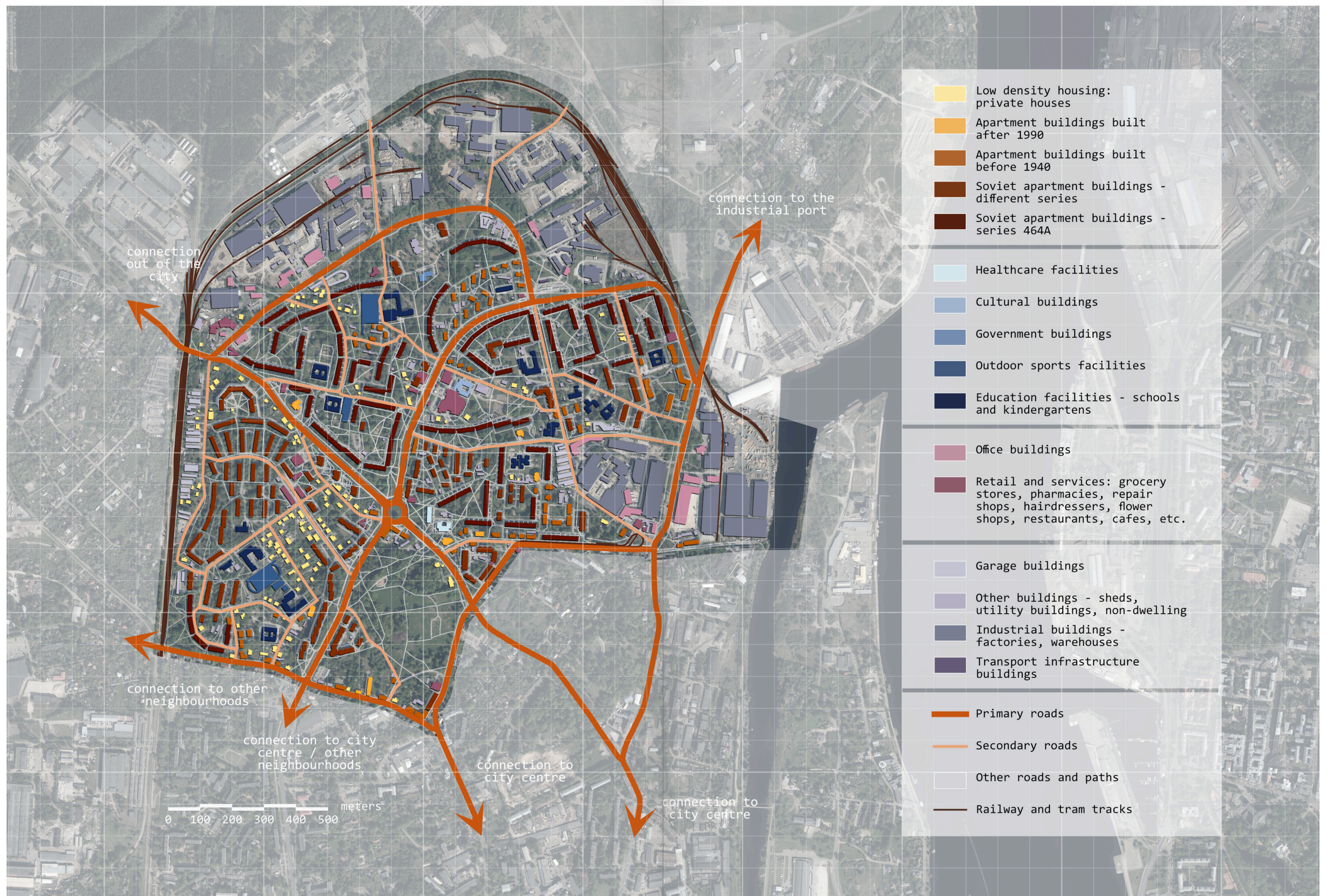


Figure 17. The comprehensive urban situation and zoning in Ilūciems

3. THEORETICAL FRAMEWORK

This section examines the foundational research and literature framing the scope of this thesis. It discusses the enduring question of the architecture of cities and the historic layers of buildings within them, rejecting the premise of architecture as a discipline that can treat a building as something entirely new each time. It argues that the alteration of existing structures within their established circumstances requires a fundamental shift in thinking, but ultimately offers vast new spatial and architectural possibilities. Furthermore, this section addresses the growing concern surrounding the linear economic model in construction and looks deeper into the circular economic approach. Finally, it explores how existing buildings can be treated as material banks instead of being viewed as static and finite entities.

3.1 THE FALLACY OF ERASURE

Architecture as a discipline has historically prioritised the creation of something new. And, of course, while a conventional architectural proposal begins with an analysis of the surroundings, most contemporary architectural practice centres around the insertion of something new into the preexisting urban fabric. It is not that the existing buildings and assemblies are disregarded completely, but the ultimate goal and ambition of an architect often culminate in the creation of something completely new, ground-breaking (figuratively and literally), and not-yet-seen. There is, of course, the discipline of conservation, restoration, and renovation, but it is viewed exactly as that – a separate discipline, rather than a core part of the architectural craft itself. However, the more centuries pass, and the more historical layers cities start to hold, the harder it becomes to pretend that we are building in an empty space, or, as defined in literature, utilising the *tabula rasa* approach.

For example, Lebbeus Woods has criticised the *tabula rasa* concept as problematic because it chooses to ignore a problem by just erasing it (Šerman & Horvat, 2023, p. 175). It is, in a way, very unnatural to erase a part of history, or, in the urban context; to tear down the buildings that are “problematic” or no longer fit for the modern “perfect” idea of how a city should be. In most other fields, society cannot just choose to erase a part of history. Of course, physical objects are more prone to decay and disappearing, but, in the modern world, it is almost impossible to erase a trace of something in its entirety. However, we as a society do tend to do that with buildings – if it is too problematic, too saturated with bad connotations, or too inconvenient, it does often become endangered.

A large portion of this modern view on architecture as a discipline stems from the architectural education - architects are mostly trained to build and design in empty spaces and land plots (Easterling, 2014, p. 7). However, the case of *tabula rasa* in modern cities could be addressed as extinct. Even if a land plot has somehow avoided being built upon throughout human history, the boundaries of architecture never end with the actual building. The context and surroundings are always there, actively affecting the chosen boundary. So, *tabula rasa*, within the

urban context at least, is an imaginary, purist concept, and utilising it is naïve and mono dimensional.

For example, in Riga, the World Trade Centre building (Figure 18), architecturally a modernist masterpiece, has been under the threat of demolition for at least the past decade (Latvijas Sabiedriskie Mediji (LSM), 2023). Why? Because of issues with its location (it is in the middle of a public park), because of ownership issues, and mainly the fact that nobody simply has any idea what to do with it next. It was built as an office building for the Central Committee of the Communist Party of Latvia in 1974 (Latvijas Sabiedriskie Mediji (LSM), 2023), during the Soviet era, therefore it does not comply with the current standards for offices. Could it then be adapted into something else instead? There have been talks on integrating it as an appendix for the ever-coming Riga concert hall (Latvijas Sabiedriskie Mediji (LSM), 2020), but that idea has been discarded as well. Adding up all these factors, it can clearly be seen that the building is simply inconvenient: too expensive to adapt, too valuable to be torn down, and too outdated to be used as it is. So, it has been standing completely empty and slowly decaying instead.

Another, very different, example is the building that used to be the KGB (State Security Committee, a repressive institution of the USSR including Soviet occupied Latvian SSR (KGB Building, n.d.)) headquarters in Riga, locally referred to as “The Corner House” (Figure 19) (Košovojs, 2014). Another architectural masterpiece, but due to the extremely negative association with this building and the horrors that have happened there in the past, it is also at risk of - not exactly demolition - but natural decay. Once again, there have been extensive discussions both on the government level and in the society on what to do with it and how to deal with that part of the history, but it is so uncomfortable, that nobody essentially wants this building. In 2024, a government-level decision was made to sell the building, simultaneously preserving the rental rights of the Occupation Museum of Latvia until at least 2050 (Latvijas Sabiedriskie Mediji (LSM), 2024), but, up until now, now no buyer has been found. It currently serves as an affiliation of the museum, dedicated to educating the public about this period of history, but the building is enormous and the museum only operates in a fraction of it. Thus,



Figure 18. The World Trade Centre building in Riga



Figure 19. The KGB building in Riga, source (Košovojs, 2014)

there it stands, in the very centre of the city, waiting for something to happen to it, but due to it being inconvenient and uncomfortable, nothing probably will. (Author's note: The destiny of this building is completely justifiable. This one specific singular building can and should be sacrificed to honour the pain and memories of several generations. due to generational trauma. However, this specific structure serves as a potent example of how spatial connotations dictate a building's chances of survival.) Architecture, through the ability to apply new connotations and functions to something preexisting, is a great tool to - maybe not overwrite history - but to turn something fundamentally tragic into something functionally valuable, turning a new page in history.

With the analysis of these examples in mind, it is very easy to conclude that tearing these buildings down would be the least-painful solution that all involved parties and society would benefit from the most. However, if we choose to do that every time something is inconvenient, a part of history is erased as well. In terms of the topic of this thesis – the Soviet-period residential buildings - they were built with a very clear goal in mind: for everybody to have housing. The goal itself has not changed, but the circumstances have. Some of the changes include evolving standards of living, legislative requirements for thermal insulation and daylighting, accessibility and fireproofing, as well as significant transformations in ways of living and family situations. But the main goal is essentially still the same: for everybody to have housing, preferably an affordable one. Therefore, it makes little sense to tear the technically “expired” buildings down because the housing is still a vital need. But it also does not make any sense to just renovate the buildings, apply a layer of thermal insulation, replace the windows, and renovate the stairwell, because that does not address the very valid concerns of having quality housing. It just becomes the same not-so-good apartment in a very nice wrapping paper. So, the question remains – can we transform the existing instead of wiping the slate clean and trying again? Pretending like a building was never there by just tearing it down is easy; working around the inconvenience of the existing is where architecture should stand (Šerman & Horvat, 2023, p. 176).

3.2 ARCHITECTURE OF DECAY

Historically, ruins and decaying structures have always been intrinsic to the nature of cities and built environments; however, how we treat these ruins has become a matter of critical importance in the past 80 years – particularly since the extreme damage inflicted by World War II (Šerman & Horvat, 2023, p. 169). The circumstances under which architecture can transform into ruins can be quite varied – from extraordinary events such as wars or natural disasters to sudden economic and political fluctuations, to the simple passage of time and the natural course of deterioration. Depending on these conditions, the process and output of the decay differ. In some cases, it is abrupt and highly visible, such as certain sections of building collapsed or damaged (mostly in cases of war or natural disasters), while in others the process is more linear and protracted, where a building simply expires and its technical condition gradually deteriorates due to lack of maintenance and repairs done on it (Šerman & Horvat, 2023, p. 170).

Naturally, decaying buildings are often perceived as something ugly and unwelcome in a built environment, but, alternatively, this process can also be viewed as a vital component of the natural life cycle of cities. It is fascinating, for example, in the case of Soviet mass housing, to see how each inhabitant has tried to improve their living conditions and spaces with their own ad-hoc methods. This is most visible in the treatment of balconies and loggias, where some residents have painted railings in bold colours, some have opted for an installation of a full-height glazing, some cultivate mini-gardens there, some use it as storage room or space for drying clothes, and others utilise it as an alternative to their fridge in the wintertime. Parallels can be drawn between this behaviour and macro-scale building decay – just as some buildings receive partial renovations (such as window replacements), while others are left untouched, balconies reflect individual timelines of use and alteration. While not strictly definable as decay, these ad-hoc interventions serve as physical evidence that the place has been inhabited and that time has passed, therefore subverting an environment that was designed to be rigidly uniform.



Figure 20. A new-built Soviet neighbourhood in Riga (*Retro Latvia Club*, n.d.)



Figure 21. A Soviet neighbourhood today (Iļģuciems, Riga)

This “decay”, or rather transformation over time, is especially important in the case of Soviet residential buildings, as they were mass-produced identically across the Soviet Union to reflect the communist ideological pinnacle of absolute equality. Consequently, exactly these ad-hoc methods of improvement and alteration were often the only way of personalisation, self-expression and spatial adaptation. Archival photographs of new-built Soviet neighbourhoods often evoke an eerie sense of absolute uniformity - everything is completely smooth, unbruised and, as intended, exactly the same (Figure 20). In contrast, walking through these neighbourhoods today reveals visible decay and deterioration, but crucially, it also reveals humans live there (Figure 21).

Rather than viewing this deterioration purely as a defect, architectural theory offers an alternative – the “romanticisation” of the ruin, transforming a defect into an effect. An “expired” building or assembly can unlock spatial possibilities that did not exist when the building was whole and functionally rigid - a ruin allows us to view a structure more authentically, revealing the dormant potential of it (Šerman & Horvat, 2023, p. 172). Within the scope of this project, the subject building is not a ruin yet, but rather a decaying structure. To prevent it from becoming an unusable ruin through the natural course of time, strategic subtraction can be employed to create an artificial ruin, and, from this manipulated state, new architectural opportunities can emerge. The advantage of utilising computational methods and a digital twin for this process is that it does not require a physical natural disaster; instead, countless scenarios of this subtraction can be simulated digitally to see the best fit option that opens up the most possibilities.

A parallel architectural strategy for addressing the ruin is the principle of superimposition: allowing the decaying structure to exist in its natural state while juxtaposing a new, contrasting element or assembly against it (Šerman & Horvat, 2023, p. 173). This approach does not treat the ruin as a flaw to be corrected, but rather as a contextual baseline that assigns value to the new architectural intervention. In the context of circularity in construction, this approach is more pragmatic, as it rejects both the zero-waste preservationist extreme and the *tabula rasa* demolition approach. It advocates for treating the existing with respect while

introducing localised new elements to elevate the performance of it.

If subtraction and superimposition are to be utilised, a critical question emerges: how to evaluate what is worth preserving and what has no value and can be removed? Can an objective system on it be established, or is this evaluation inherently subjective? (Šerman & Horvat, 2023, p. 177) Evaluating these questions in the context of this project, the answer relies on physical objectivity: a built assembly is fundamentally a physics-based system, therefore obeys the laws of gravity and civil engineering. Therefore, each element possesses a quantifiable role in an existing assembly - it is either load bearing or it is not. While the actual relationships between the elements of the assembly are far more complex (as will be mapped out in subsequent chapters), understanding what is structurally vital allows for a clear, objective categorisation of elements within a singular building.

As Keller Easterling suggests, since architects are trained in the erection and making of a building, they should also be the ones to possess the knowledge about the unmaking of it. Understanding how a building works inherently should also mean the understanding of exactly which subtraction will cause the system to fail and stop functioning (Easterling, 2014, p. 4). This approach of controlled unmaking inevitably raises the question of authorship - an author of a creation also holds the ability to destroy said creation (Easterling, 2014, p. 6), but what happens when an architect deconstructs the work of another, especially if the original creator is long gone? And the question becomes even more complex in the Soviet architecture and construction scope, because singular authorship over a design was systemically erased by the state, therefore it does remain unknown in most cases. Because singular authorship was not allowed to exist, these buildings belong less to and architect and more to the collective timeline of the city and history, justifying their contemporary architectural unmaking.

3.3 THE CIRCULAR SHIFT

The relevance of material reuse, repurposing and reclaiming is as high as ever (Önalan et al., 2025, p. 1). The need for new,

raw materials in the construction industry has long surpassed the natural regeneration capabilities of the Earth, as the sector is already consuming 50% more natural resources than it did only 30 years ago (Tzourmakliotou, 2021, p. 1). This overconsumption of materials and resources is a result of the linear economic model, driven by the “take – make – dispose” approach (Tzourmakliotou, 2021, p. 1), in which the core ideas centre around a single life cycle of an element, in this case, a building. When the life cycle of this building comes to an end, the materials used are disposed of, and a new cycle begins, with no attachment or connection to the previous one. In response to the alarming tendency of diminishing resources, the governments of the world have started to implement stricter environmental policies to protect the environment, thus largely impacting the processes in the construction industry (Tzourmakliotou, 2021, p. 2). The main reason behind construction material and element reuse not being a common practice yet is that current construction processes are designed to serve the classic linear “take-make-dispose” economic model from both a legislative and a design framework (Kloft et al., 2024, p. 106).

As defined by Max Eschenbach in his 2023 publication, two types of process chains can be identified. The conventional architectural discipline and construction processes operate on digital-real process chain, where building components and elements are digitally designed and made through the consumption of raw materials. Within this approach, many aspects can be optimised to the maximum extent and large amounts of individualised components can be produced. However, even though this maximal optimisation can save some resources, it is diminished by the fact that raw materials are still used in production, alongside very complex assembly procedures and the fact that components are becoming increasingly over-individualised which prevents further element reuse (Eschenbach et al., 2023, p. 443). From this approach a question emerges: even though in the modern construction industry additive manufacturing is taking on an increasingly significant role, which means that all components can be individualised, is it sustainable? Looking at this from the perspective of reuse possibilities, more uniform and standardised elements are more likely to get a second

chance. Just as Keller Easterling states in the context of whole buildings, where the more mono-functional a building is, the more at risk for deletion or demolition it is as soon as it is built, (Easterling, 2014, p. 26), the same ideology can be applied on an element scale. The more individualised and specific the original building element is, the less likely it is to be reused in the next life cycle.

Since the linear economic model no longer conforms to current needs and concerns, it has exhausted its capabilities and a new circular economic approach must be implemented (Zboinska & Göbel, 2025, p. 13). This approach implies that the entire life cycle of a material should be determined the moment it is included into a design, predicting and planning its possible reuse in future construction cycles (Önalan et al., 2025, p. 2). Implementing this idea into practice, however, comes with its limitations and challenges, the most severe being the significant shift in the design process itself, transitioning from the idea of “form follows function” to that of “form follows availability” (Önalan et al., 2025, p. 2). The material can no longer be made to fit the design intent, but rather the opposite – the project has to be fit around the material that is available, therefore shaping the architecture and form. Here the real-digital process chain as defined by Max Eschenbach comes in. The process is dictated by a physical building element that has already been materialised in a component with certain geometry and properties. These can then be digitalised and recombined with the use of computational tools (Eschenbach et al., 2023, p. 443). In this approach, another aspect is crucial: the idea of reconditioning elements that are no longer needed into something that is needed for the design or building (Kloft et al., 2024, p. 108).

This shift and reversal of process chains and workflows, working with repurposed elements and designing for disassembly, may be the next defining chapter for modern architecture. It is, in a way, a new discipline altogether as it requires new ways of thinking and new skills (Zboinska & Göbel, 2025, p. 1), as well as setting aside personal ambition and ego, since the element an architect is working with may have been designed by someone else before.

The primary aspects and goals of design within the circular economic approach are closing material loops by integrating material

availability into the design and construction processes, optimising the use of resources, minimising waste, and extending the life cycle of materials (Önalán et al., 2025, p. 2). Furthermore, even though new tactics often tend to have an “all-or-nothing” approach - where either every part of a building has to be sustainable, locally sourced, reused and reclaimed, or nothing at all - embedded in them, when working with reclaimed materials, one or a few relevant objectives and goals should be picked rather than trying to fit all of them (Önalán et al., 2025, p. 2). Currently, mainly non-load-bearing elements and materials are considered in the scope of reuse, but carbon is mostly embedded in load-bearing elements, therefore making them the most important to be reused. For now, these elements are usually approached as one-life-cycle elements only and discarded afterwards (Kloft et al., 2024, p. 106). The constraints set up by this “all-or-nothing” approach can easily contradict each other, often terminating any further process development. Therefore, to physically implement material and component reclamation into real-world projects, realistic goals and objectives should be set. The goal, therefore, should not be to reuse each and every element, because there will always be a percentage of them too damaged or too specific to be targeted for reuse, but instead to start reusing at all (Kloft et al., 2024, p. 113).

3.4 BUILDINGS AS MATERIAL BANKS

The traditional notion of a building carries the idea that it is a final form, created for a specific purpose and intended to serve for a limited amount of time. After this life cycle, the building is torn down, the materials are discarded, and a new building with a new function is built from new elements. In some cases, such as vernacular construction, this does make sense, as the materials are natural and are severely affected by the forces of nature and weathering, and essentially rot and become compost again. However, the scope of this thesis examines prefabricated concrete assemblies, and the case for concrete elements is vastly different. As discussed by Harald Kloft (2024), concrete itself does not lose its strength unless it is subjected to chemical damage. The reinforcement, however, does – it is affected by degradation, corrosion, and rust, especially if the elements are exposed to

moisture in prolonged periods of time (Kloft et al., 2024, p. 112). This means that the viability of reusing precast elements essentially comes down to being able to evaluate the reinforcement within them.

How, then, should buildings be approached as material banks? The process starts with finding a donor structure (Kloft et al., 2024, p. 109) – a structure that has served its intended lifespan and can no longer be used for its intended purpose, and therefore has become abandoned or unexploitable. Within this donor structure, individual elements can then start to be evaluated. Each element can be strength-tested, and the weaker elements that do not pass the tests can be strengthened (Kloft et al., 2024, p. 110). However, as discussed previously, not every element can be subject to reuse, and the deemed unsuitable can instead be used to inspect other similar components. These discardable elements can then be examined by taking them apart, subjecting them to excessive strength testing that may damage the already unusable component, or dismantling them to test the reinforcement and other properties (Eschenbach et al., 2023, p. 446). This approach can help resolve the dilemma of how to detect and test the reinforcement within reinforced concrete elements, because reliance on historic construction drawings can often be inaccurate.

There is another similar approach, defined by Brandon Clifford as cannibalism in architecture: “[...] it references inanimate matter consuming itself, or its own likeness. This cannibalism is an inhuman one, in which a material sameness consumes its own likeness” (Clifford et al., 2018, p. 587). In his research, he states that cannibalism, while seemingly a modern and novel approach, is in fact an ancient practice. Earlier cultures often took apart buildings and monuments created by their predecessors to use the materials in new assemblies (Clifford et al., 2018, p. 588), therefore categorising this method as a natural course of the building life-cycle that the modern construction industry seems to have forgotten.

A very interesting example, directly applicable to the scope of this thesis, is a new apartment building construction currently ongoing in the analysed neighbourhood of Iļģuciems (Figure 22). It can be observed clearly that this new building is being



Figure 22. A new construction currently ongoing in Iļģuciems, with series 1-464A residential buildings in the background

constructed from prefabricated concrete panels. Upon closer inspection of the project, the exterior walls are thermally insulated 3-layer concrete panels with an exposed concrete finish (Aventino, n.d.). What is interesting about this project is that it is located in a plot surrounded by Soviet residential buildings, and more specifically, exactly the object of this research – series 1-464A. Even though the elements differ slightly, it makes for a very surreal case study and example: there are buildings essentially falling apart and in danger of being evacuated because of “expiration” right next to this new-build, which is using the exact same construction methodology, materials, components, and even architectural aesthetic. It is, in a way, completely absurd, and would make for the perfect reuse case – the donor is right there. Of course, the elements themselves would require adjustments both in them and the planned architecture, but instead, raw materials are used to repeat this same linear deterioration and disposal approach in another 70 years.

However, both the approach of using a donor structure and cannibalism in literature mostly speak about having the donor separate from the receiver. Within the scope of this thesis, a methodology that could be defined as “self-cannibalism” is proposed. The building itself is treated as a donor, but it

is the receiver at the same time. The elements subtracted from the building to create new spatial arrangements and layouts are re-implemented back into the building afterwards. As discussed previously, this is not a zero-waste method, as it relies on many factors both computationally and spatially, and not all elements can or should be considered for reuse. The strength of the methodology lies within the site-centric approach, where the reusable materials do not have to be transported. Another strength of this approach is using the existing structure with existing structural foundations and relations, eliminating a very resource-intensive factor that is impossible to avoid in the classic donor-receiver relationship model. This approach is also supported by Kloft, who implies that the reapplication of load-bearing elements in different assemblies should be executed only if the original load-bearing structure itself cannot be preserved (Kloft et al., 2024, p. 106). In this case, the load-bearing structure of the 1-464A series buildings is structurally fairly safe (Dzirkalis et al., 2019), and therefore the approach chosen focuses on reorganising the existing structure without disrupting or worsening it, while ensuring element stability. As defined by Kloft, preserving load-bearing structures is the easiest, most crucial step towards sustainability in construction (Kloft et al., 2024, p. 106).

4. TYPOLOGICAL DIAGNOSIS: 1-464A SERIES

For the scope of this research, the 1-464A series was selected because it is typologically prevalent in the focus neighbourhood of Iļģuciems, critically endangered by structural “expiration”, and possesses comprehensive technical and archival documentation. This chapter investigates both the structural and spatial anatomy of the building series. The structural scheme is evaluated by studying technical inspection reports alongside archival construction drawings and montage schemes, exposing the physical vulnerabilities of the assembly. Concurrently, the architectural anatomy is assessed and analysed through the prism of modern European legislative documentation to objectively diagnose its spatial obsolescence.

4.1 TECHNICAL ANATOMY

In order to gain a comprehensive understanding of the series 1-464A buildings and their construction anatomy, several technical inspection reports, carried out in 2019 by “CMB” as an inquiry by The Ministry of Economics of Latvia, were inspected and analysed.

Building element	Material and structural composition	Technical wear	Main issues and causes	Digital model designation	Identifier
Foundations and base	Sub-surface part: assumed to consist of reinforced concrete pile foundations with strip-type lattice with an accessible basement (Lejiņš et al., 2019, p. 5).	25%	Local leaching and groundwater infiltration. Accumulation of moisture due to failing/outdated waterproofing layers and lack of vertical drainage for groundwater (Lejiņš et al., 2019, p. 6).	Basement modelled as monolithic and non-alterable	N/A
Above-ground basement walls	Composite panels consisting of two external reinforced concrete slabs with an aerated concrete insulation layer (Lejiņš et al., 2019, p. 6; Dzirkalis et al., 2019, p. 16).	30%	Subject to lateral pressure from surrounding soil if located below ground level (Lejiņš et al., 2019, p. 7; Dzirkalis et al., 2019, p. 16).	Modelled as panel elements	0x
Above-ground external load-bearing walls	Precast expanded clay aggregate concrete panels (approx. 300mm thick) finished with decorative plaster (Lejiņš et al., 2019, p. 6; Dzirkalis et al., 2019, p. 16).	30%	- Insufficient concrete cover of welded steel nodes allows moisture ingress, triggering oxidation, rust, and cracking of the concrete cover (Strazdiņš et al., 2019, pp. 20–21). - "Imperfect" panel alignment creates eccentricity that reduces compressive strength (Strazdiņš et al., 2019, pp. 20–21). - The misalignment of essentially makes the building assembly “act” as a masonry wall, making each panel act as a “brick” in this assembly (Strazdiņš et al., 2019, p. 20).	Modelled as panel elements	1x (exterior wall panels for floors 1-5) 2x (exterior wall panels for roof / parapet)
Above-ground internal load-bearing walls	Solid reinforced concrete panels with a thickness of 120mm – 150mm (Lejiņš et al., 2019, p. 6; Dzirkalis et al., 2019, p. 16).	30%	Same as external walls (corrosion of embedded steel parts and loss of compressive strength due to imperfect alignment).	Modelled as panel elements	ISx (transverse wall panels) ISGx (longitudinal wall panels) ISRx (secondary non-load-bearing wall panels) KTx (staircase wall panels)

Table 2. Structural elements of the series 1-464A, their material composition, main issues and causes, and created model designation

Building element	Material and structural composition	Technical wear	Main issues and causes	Digital model designation	Identifier
Floor slabs (interfloor, basement, attic)	100mm thick reinforced concrete prefabricated panels, supported along their perimeter on the load-bearing walls (Dzirkalis et al., 2019, p. 32).	30%	Original design assumes four-sided support, but inaccuracies/settlement mean slabs are not supported equally on all sides. Slabs designed for "plate behaviour" are forced to act as beams, creating unintended stresses and cracking (Dzirkalis et al., 2019, p. 32).	Modelled as panel elements	FPx (floor slab panels)
Joint sealing, waterproofing, and thermal insulation	Joints: Welded via U-shaped clamps/flat bars (Lejiņš et al., 2019, p. 6) and filled with dolomite aggregate concrete grout (Dzirkalis et al., 2019, p. 22). Waterproofing: Bitumen mastic in two layers for basement exterior (Dzirkalis et al., 2019, p. 29). Thermal insulation: Minimal/absent (Dzirkalis et al., 2019, p. 29).	N/A	Severe thermal bridging and moisture ingress. Fills and cement mortar degrade over time; expansion and contraction cycles pull layers apart, creating non-hermetic openings. Does not comply with thermal regulations LBN 002-15 (Dzirkalis et al., 2019, p. 29).	N/A	N/A
Roof and spatial stability elements	Gable roof structure with internal rainwater drainage system, the roof slope is directed towards the central longitudinal axis of the building. Reinforced concrete prefabricated beams resting on walls, supporting floor slabs and roofing material (Lejiņš et al., 2019, p. 15).	N/A	Spatial stability relies entirely on load-bearing walls and panel ceilings; no additional stability elements exist (Lejiņš et al., 2019, p. 15).	For the simplicity, roof structure is treated the same as the interfloor slabs	FPx (floor slab panels)
Balconies, loggias and entrance roofs	Loggias: Slab panels supported by exterior and partition walls (Lejiņš et al., 2019, p. 17). Entrance roofs: Concrete panels resting on walls and external steel round pipes (Lejiņš et al., 2019, p. 18).	30%	Peeling of the originally thin (100mm) protective concrete layer due to harsh weather and freezing cycles, exposing reinforcement rebars to the atmosphere (Lejiņš et al., 2019, p. 17).	Modelled as panel elements	BPx (loggia slabs) BRx (loggia railings) EPx (entrance panel elements) ESx (entrance roof and slab elements)
Stairs and ramps	Reinforced concrete prefabricated staircases and landings resting on load-bearing walls, equipped with reinforced metal railings (Lejiņš et al., 2019, p. 19).	20%	Standard technical depreciation.	Modelled as solid elements	STx (stair runs) SLx (stair landings)

4.2 STRUCTURAL VULNERABILITIES

To understand the structural vulnerabilities of the assembly, the material and structural behaviour of the elements of the assembly must be understood. The 1-464A series building is characterised by a significant weight disparity: the internal reinforced concrete wall and floor panels are extremely dense (~2200kg/m3), while the external expanded clay concrete panels are relatively light (~1050kg/m3) (Strazdiņš et al., 2019, p. 11).

The structural integrity of this assembly relies entirely on its joints – the wall joints exposed within the technical inspection reveal that the reinforcement elements are welded together and embedded in dense concrete mortar, creating rigid connections (Strazdiņš et al., 2019, p. 13). Because of this rigid, stacked dependency, structural evaluations conclude that the 1-464A series building assembly fundamentally functions as a masonry building. However, instead of small bricks as it is in the case of typical masonry wall, each massive precast wall panel acts as a singular, independent load-bearing block (Strazdiņš et al., 2019, p. 20).

This “macro-masonry” behaviour exposes the key vulnerability of the building: lack of systemic robustness. Because there is no continuous mutual anchoring reinforcement running through the interpanel joints, the building is not protected against progressive collapse. A progressive collapse is defined as the collapse of a structure or a significant part of it, caused by a collapse of a singular, localised element or a part of it, triggering a chain reaction that compromises the stability of the entire structure (Strazdiņš et al., 2019, pp. 20–21).

According to technical inspections, there are two main factors that can trigger a progressive collapse in the 1-464A series buildings. The first is environmental, where it can be triggered by excessive deformation within the structure of foundations, caused by prolonged rainwater leeching into the shallow soils. The second, and most critical to the scope of this thesis, is human intervention: arbitrary or ill-considered partial or complete dismantling of load-bearing walls during unpermitted spatial redevelopments (Strazdiņš et al., 2019, p. 21). Because each panel of the assembly relies directly on the

one beneath it, removing a singular wall without systemic calculations risks the stability of the entire structure.

Overall, technical calculations indicate that the steel connecting elements currently possess sufficient load-bearing capacity. Provided that the joints remain hermetic and free of severe corrosion, the 1-464A buildings are technically safe for continued operation in their current, unaltered state (Strazdiņš et al., 2019, p. 21).

Technical reports explicitly state that improving the robustness of these buildings through traditional reconstruction methods, such as monolithizing of the walls and introducing new interpanel anchorages, is economically and technologically inefficient (Strazdiņš et al., 2019, p. 21). This conclusion is incredibly significant for the future of Soviet mass housing, as it dictates that brute-force structural retrofitting is a dead end. Therefore, if the obsolete spatial layouts of these buildings can be transformed, it cannot be done through arbitrary manual demolition or expensive local monolithic reinforcement. It necessitates a highly calculated, precise methodology of the computational subtraction methodology proposed in this research.

4.3 SPATIAL AND ARCHITECTURAL ANATOMY

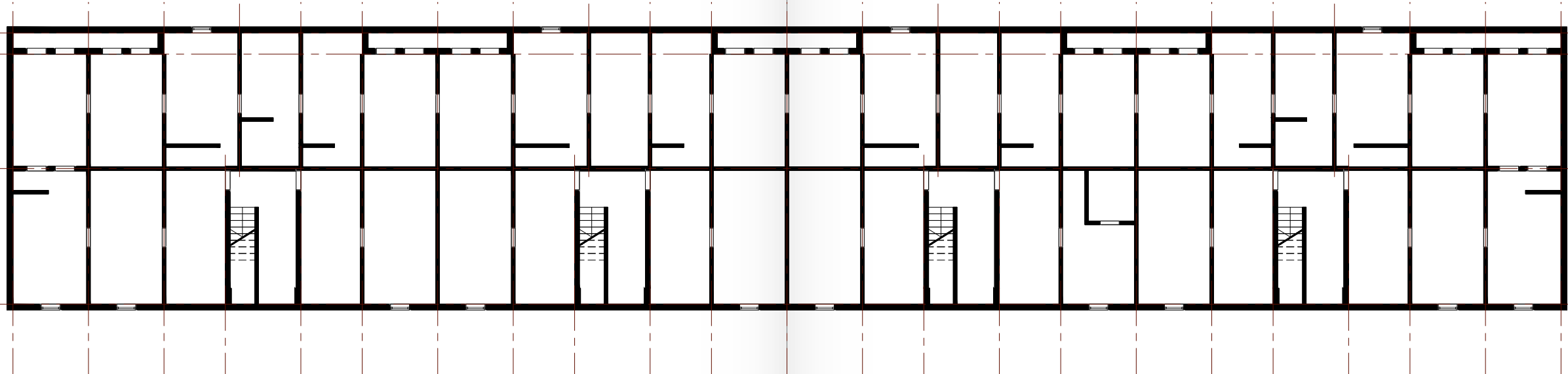
The architectural anatomy of the building follows the strictly standardised Soviet mass residential housing construction ideology – the floor plans (Figure 23) are as standardised as possible. The building comprises five floors, and the layout is identical across all levels. It contains 60 apartments in total, distributed as 12 apartments per floor and 3 apartments per stair landing within each vertical circulation core. The building consists of 4 distinct sections served by 4 staircases. Excluding mirrored variations, there are 6 primary apartment layout types within this building. The specific building chosen for this thesis consists of 2 end modules and 2 centre modules. However, in different variations of the 1-464A series, these modules can be recombined and expanded: while a building will always terminate with 2 end modules, it can theoretically accommodate an infinite number of middle modules between them.

On each floor, there are 3 types of 2-bedroom apartments (6 total), 2 types of 1-bedroom apartments (4 total), and 1 type of studio apartments (2 total). Each apartment type, apart from the studio apartments, features a private outdoor space – a loggia. Each apartment has one bathroom block – in all types, the bathroom is separated from the toilet, except in the studio apartments where they are combined. All floors share identical layouts except for the ground floor, where the entrance zones to the staircases protrude outwards as separate functional volumes. The building also includes a basement, accessible from each entrance node via a staircase. This basement is not inhabitable and is currently used as a storage and technical service space. Since the ceiling height of the basement is very low (2.1m), it is not considered a viable space for architectural interventions within the scope of this thesis. The standard ceiling height within the apartment spaces is 2.6m.

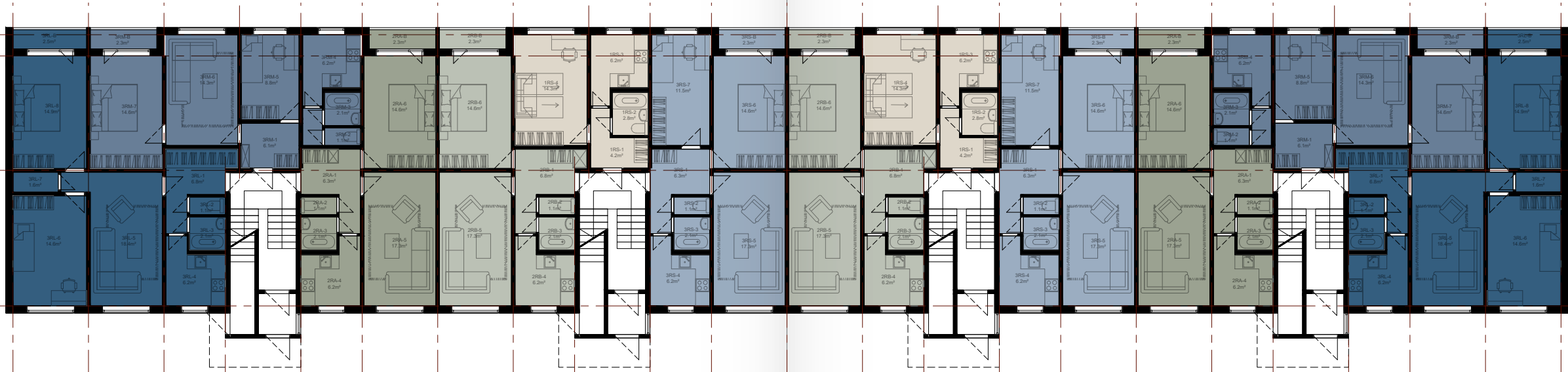
Apartment type	Entrance (m²)	Bathroom + WC (m²)	Kitchen (m²)	Living room (m²)	Bedroom 1 (m²)	Bedroom 2 (m²)	Other (m²)	Loggia (m²)	TOTAL (m²)
3RL	6.8	3.2	6.2	18.4	14.6	14.9	1.6	2.5	68.2
3RM	6.1	3.2	6.2	14.3	14.6	8.8	N/A	2.3	55.5
3RS	6.3	3.2	6.2	17.3	14.6	11.5	N/A	2.3	61.4
2RA	6.3	3.2	6.2	17.3	14.6	N/A	N/A	2.3	49.9
2RB	6.8	3.2	6.2	17.3	14.6	N/A	N/A	2.3	50.4
1RS	4.2	2.8	6.2	14.3	N/A	N/A	N/A	N/A	27.5

Table 3. Room area comparison between the apartment types in the building

BASEMENT FLOOR



GROUND FLOOR



STANDARD FLOOR

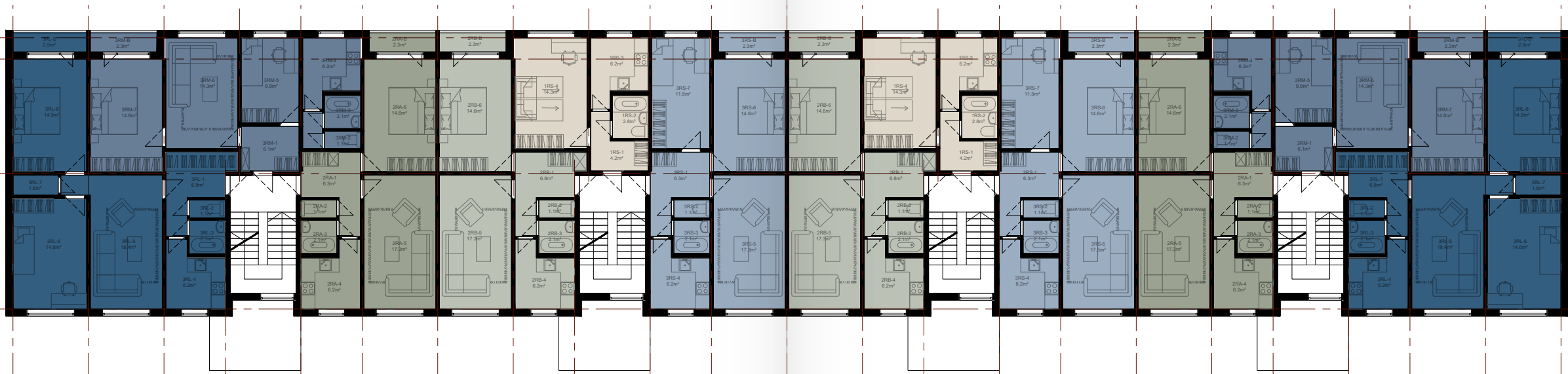


Figure 23. Original floor plans of the object building of the 1-464A series

4.4 SPATIAL DEFICIENCIES

To contextualise the research relevance within the current architectural environment, an in-depth analysis of these layouts was conducted. Since Latvia does not currently enforce strict minimum space requirements for apartments, the evaluation of the series 1-464A is benchmarked against modern European standards for dwellings. For this analysis, the spatial organisations are critiqued against the UK's Nationally Described Space Standard (Department for Communities and Local Government, 2015) for overall area metrics, and the Swedish Boverket Building Regulations (Boverket, 2019) for modern functional accessibility.

In terms of overall living space metrics, the UK standard dictates that a 1-bedroom apartment for one person should be at least 37 m², and at least 50 m² if inhabited by two people (Department for Communities and Local Government, 2015). In contrast, the 1RS studio apartment in the object building is only 27.5 m², despite the assumption that in most cases at least two people live in it. The situation is marginally better in the case of the full 1-bedroom apartments with a separate living room (2RA and 2RB), where the actual area of 49.9 m² and 50.4 m² aligns closely with the 50 m² minimum (Department for Communities and Local Government, 2015). Similarly, for the 2-bedroom apartments (3RS, 3RM and 3RL), the standard is partially met if occupied by three people (requiring 61 m²) as the actual area ranges from 55.5 m² to 68.2 m². However, in a case more realistic for Riga, where four people live in this apartment (for example, parents and two children), the standard is substantially failed, as the UK regulations require at least 70m² for this occupancy level (Department for Communities and Local Government, 2015).

A critical deficiency applicable to almost any Soviet residential building, including the 1-464A series, is universal accessibility. The Swedish Boverket requires that the entire ground floor be fully accessible to wheelchair users (Boverket, 2019, p. 22). This requirement is entirely unmet, as the ground floor is elevated one flight of stairs above street level, and there are no elevators or chair lifts in the stairwells. Furthermore, while the stair landings are spatially sufficient for a wheelchair to manoeuvre, the entrance doors to the apartments are not wide enough for an accessible entrance and feature a noticeable

threshold that worsens the accessibility even more. Consequently, since even the public and semi-public spaces are inaccessible, the private interiors are inherently worse. This is a crucial issue, because it affects not only the wheelchair users, but also the elderly, people with movement limitations, and families with small children: the spaces are cramped, fragmented, the openings are small and the movement within the apartments can be very inefficient and difficult.

The next critique is directed at the sanitary blocks. While the separation between a bathroom and a toilet can be a convenient spatial arrangement, in this particular case it must be evaluated as ineffective because both spaces are cramped, and there is not even room for a washbasin in the toilet room. Additionally, this arrangement entirely fails to comply with any modern accessibility requirements, as there is no possibility to manoeuvre a wheelchair within spaces that feel too narrow even for able-bodied users to walk into.

A further critique addresses the kitchens. As mentioned previously, they were kept small in Soviet projects to minimise private gatherings and encourage social mingling outside of the homes. However, in the modern living context, this presents two major issues. Firstly, the kitchen size is exactly the same in every apartment type - no sizing differences exist between a studio apartment suitable for one person or a 2-bedroom apartment possibly housing a family of four. While the space required for cooking and appliances does not differ drastically depending on the amount of people living in an apartment, the dining space should, and that is simply not possible under these circumstances. In modern culture, the kitchen is more than a space for cooking - it also functions as a gathering and social space, making the existing arrangement outdated. Secondly, while the dimensions technically comply with the minimum requirement that the space for a cooking surface should be at least 1.8 m wide in dwellings smaller than 35 m² (Boverket, 2019, p. 24), the requirements for accessibility, adequate space for modern appliances, and a dining area near the cooking zone are completely unmet.

Finally, the floor plans suffer from severe spatial fragmentation. Since the rooms are quite small, the layout acts as a spatial labyrinth with small, often dark, cells. And



Figure 24. Cramped interior spaces typical of a 1-464A series residential building

while the daylighting conditions of the apartments are quite optimal - no matter how the building is oriented on the site, because most of the apartments (around two-thirds) have windows on both sides of the building - the spaces towards the core of the building (the corridors, bathrooms and entrance areas) remain quite dim and therefore spatially oppressive. Another outdated concept seen in these layouts is the walk-through living room. While this technically allows for maximisation of "living" space by abandoning redundant corridors, it limits the possibility to both utilise the walk-through space effectively and use the living room for a different function (for example, as another bedroom).

Overall, this analysis clearly shows that the layouts in the 1-464A buildings are outdated and unsuitable for modern living, even though the technical condition of the building allows it to be kept inhabited. Therefore, a system designed to address both the structural weaknesses, and the spatial deficiencies must be created.

5. THE COMPUTATIONAL METHODOLOGY

This chapter scales down to examine the building at the modular prefabricated element level. It discusses the creation of a high-reliability digital twin and element catalogue, as well as the evaluation of the micro-scale inter-element relationships. It also explores the macro-scale methodological framework for the alteration algorithm and discusses the issues identified during this process. This section essentially functions as a computational preamble for the specific alterations defined in the subsequent chapter.

5.1 DIGITAL CATALOGUING

To develop a system of alteration, a high-reliability digital twin must first be created. An inventory survey of a specific 1-464A building within the research neighbourhood of Ilguciems was chosen as a base reference for the overall structural dimensions. To determine the exact dimensions of the precast concrete panels, both technical inspection documents and archival construction drawings were utilised. While the physical building is affected by deviations and imprecisions due to natural structural settlement and wear, this project assumes that all originally identical elements remain geometrically uniform to facilitate subsequent algorithmic processing.

Initially, the inventory documentation was translated into a two-dimensional CAD framework to map the plans and layouts analysed in previous chapters. Then, using these plan drawings in combination with visual documentation (images), the external wall panel layout of the facades was outlined. Subsequently, the preliminary exterior panel dimensions were cross-referenced with the plan dimensions, and the structural grid was adjusted to minimise the count of unique elements and maximise geometric uniformity, reflecting the original construction intent. Next, based on this refined grid, the exact locations and dimensions of the interior wall panels were mapped out, producing a finalised iteration of the floor plans. Panel openings (windows and doors) were then extracted from both the inventory documents and archival construction drawings. Finally, the horizontal elements - the floor slabs - were identified based on the grid created, again minimising the unique element count.

The next phase of the digital twin creation involved transforming the two-dimensional CAD framework into a three-dimensional computational model in Rhinoceros 3D. The two-dimensional geometries were extruded according to the specifications analysed in Section 4.1. Crucially, all unique elements were instantiated as block references to ensure strict geometric identity for the subsequent computational steps. Each element was assigned a unique identifier based on their structural role and typology (as detailed in Table 2 in Section 4.1). The basement and sub-surface foundations were modelled separately as a singular solid mass, as they are assumed to consist of cast-in-situ

monolithic concrete, therefore remain unalterable in the scope of this thesis.

Once the comprehensive building model was completed, the three-dimensional element catalogue was compiled, isolating every unique precast component. The element catalogue can be seen in Figure 26, while the elements contextualised within the full assembly are illustrated in Figure 25.

Although the framework built for alterations of the structure relies on the modularity and repetition of identical elements, addressing the local and site-specific individualities requires a more comprehensive survey of the physical building. Therefore, while this thesis utilises manual translation from archival documents and inventory surveying, in an ideal, real-world application, the creation of the digital twin would integrate high-resolution photogrammetry and point-cloud scanning to capture the precise as-built conditions of the existing structure.

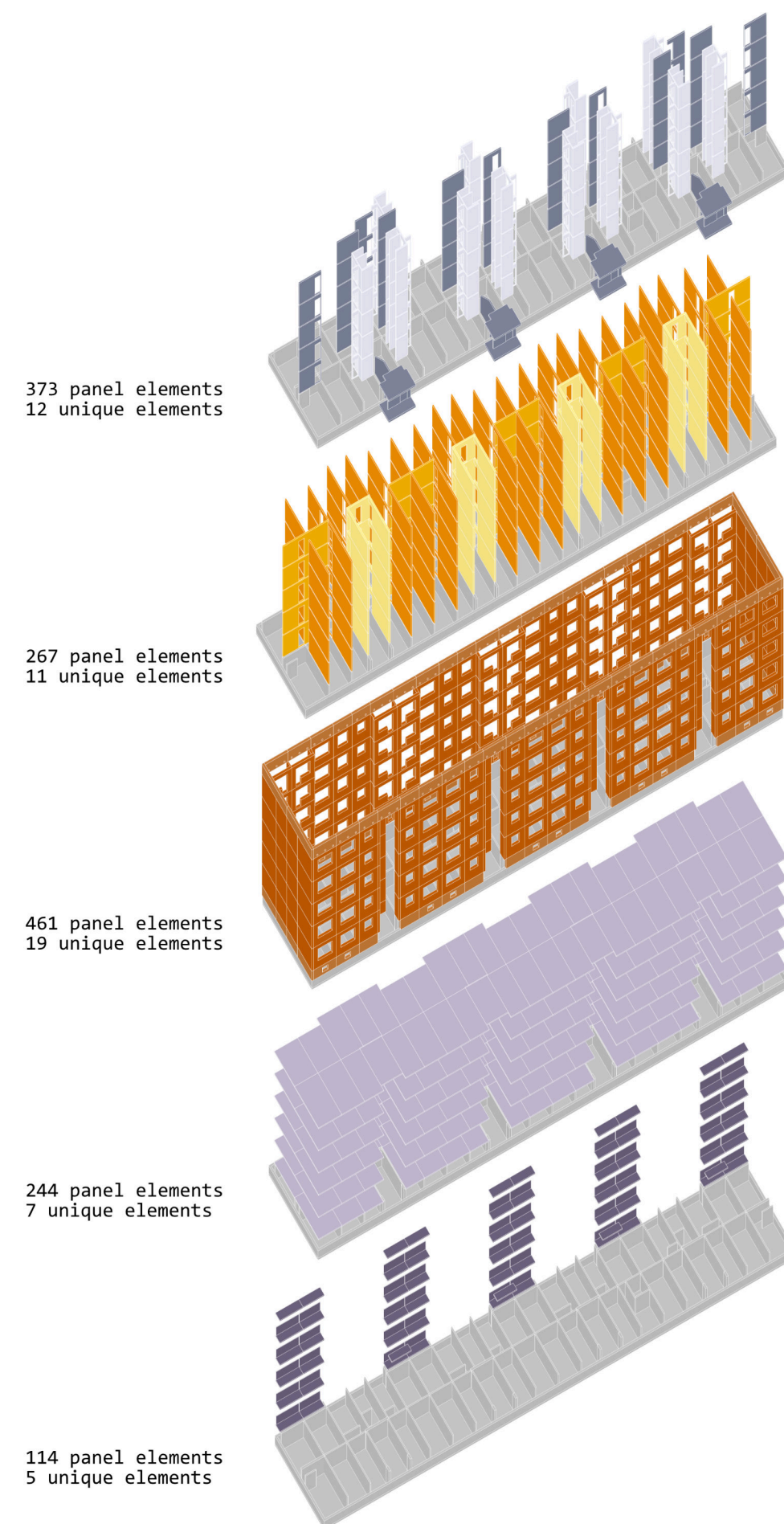


Figure 25. Element catalogue in the context of the overall building assembly

external load-bearing wall panels	1A  70pcs	1C  60pcs	1D  20pcs	1E  25pcs	1E_M  25pcs	1F  16pcs	1F_0  4pcs	1G  40pcs	1H  25pcs	1J  40pcs	1K  20pcs	1L  20pcs		
external load-bearing base panels	0A  14pcs	0C  20pcs	0E  10pcs	0L  4pcs	external parapet panels	2A  28pcs	2C  16pcs	2B  4pcs						
transverse internal load-bearing wall panels	IS1  25pcs	IS2  10pcs	IS3  20pcs	IS4  35pcs	IS5  30pcs	IS6  40pcs	IS7  10pcs	longitudina l internal load- bearing wall panels	ISG1  10pcs	ISG2  15pcs	internal stairwell load-bearing wall panels	KT1  24pcs	KT2  48pcs	
internal non load-bearing wall panels	ISR1  10pcs	ISR2  20pcs	ISR3  10pcs	ISR4  10pcs	ISR5  10pcs	internal bathroom blocks	WC1  40pcs	WC2  10pcs	WC3  10pcs					
floor slabs	FP1  12pcs	FP2  12pcs	FP3  48pcs	FP4  72pcs	FP5  48pcs	FP6  48pcs	FP7  4pcs	loggia floor slabs	BP1  12pcs	BP2  48pcs	BP3  4pcs			
stair elements	ST1  32pcs	ST2  4pcs	ST3  4pcs	SL1  20pcs	SL2  16pcs	loggia railing panels	BR1  10pcs	BR2  40pcs						
entrance block wall elements	EP1  4pcs	EP2  4pcs	EP3  4pcs	entrance block slab elements	ES1  4pcs	ES2  4pcs	ES3  4pcs							

Figure 26. Individual element catalogue detailing unique precast components

5.2 ELEMENT DEPENDENCY DEFINITION

Once all unique elements were accounted for, the subsequent step was to establish their physical macro- and micro-scale relationships. Although the primary structural role (load-bearing versus non-load-bearing) of each element was identified in the previous phase, the localised inter-element dependencies remained unclear. Since this thesis proposes a perspective partial disassembly of the structure, an assembly sequence (Figure 28) was reconstructed based on archival montage drawings. This step was crucial in visualising the spatial and structural relationships between the elements.

Following this, a micro-scale dependency matrix was constructed. This was achieved through digital collision and adjacency detection to identify all unique instances of one element touching another. This process was executed across sets of element groups, for example, all exterior wall panels were tested against interior load-bearing wall panels, as it can be clearly observed in the assembly sequence that these two element typologies provide mutual lateral bracing for one another. To ensure a comprehensive mapping of all dependencies and structural relationship, these adjacency tests were systematically run across every possible pair of element categories. This adjacency detection was done via a custom Grasshopper script. The output of this step was a CSV data matrix showing every physical contact between element pairs, with all duplicate intersections eliminated throughout the process (Figure 29).

Because the output of the previous step consisted of raw geometric adjacency data, specific structural relationships had to be assigned to the element pairs manually. Each pair was assigned a relationship type and a structural hierarchy level. These variables are explained in Table 4 and Table 5.

During the process of the creation of this dependency matrix, it became evident that the load paths within the assembly are not strictly linear or vertical but rather operate in a reverse-triangle load distribution. Once fully established, this structural matrix served as the foundational logic for the next phase: translating the raw CSV dependency data back into the Grasshopper environment.

Relationship type	Description
Vertical stack	The dependent element (wall panel) is stacked directly above the support element (wall panel).
Horizontal Support (FPx)	The dependent element (floor slab panel) is supported by the support element (wall panel).
Mutual lateral bracing	The dependent element (wall panel) is laterally and perpendicularly braced by the support element (wall panel). The bracing is mutual.
Lateral bracing	The dependent element (wall panel) laterally and perpendicularly braced by the support element (wall panel). The relationship is unilateral.
Mutual parallel bracing	The dependent element (wall panel) is parallelly braced by the support element (wall panel). The bracing is mutual.
Floor supports the wall	The support element (floor slab panel) supports the dependent element (wall panel) positioned atop it.

Table 4. Relationship types between building assembly elements

Structural hierarchy	Description
Primary	Applied to all vertical stacks of elements. The removal of the support element catalyses catastrophic progressive collapse.
Secondary	Applied to horizontal supports and most bracing relationships. The removal of the support element can cause the localised collapse of the dependent element.
Tertiary	Applied to specific bracing relationships involving non-load-bearing elements. The removal of the support element may induce instability but does not trigger immediate collapse.

Table 5. Structural hierarchy between building assembly elements

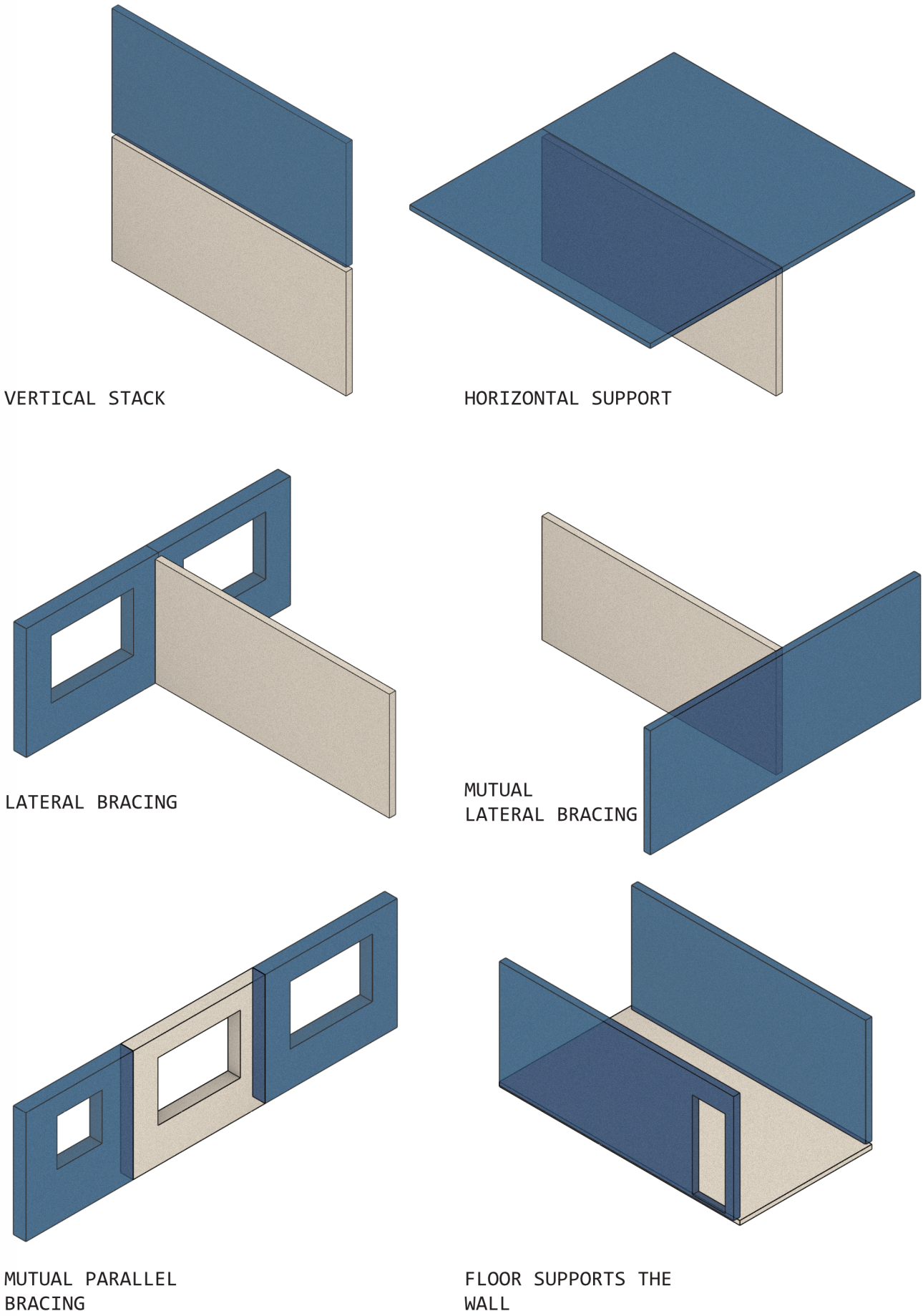


Figure 27. Inter-element relationship examples highlighting structural dependencies

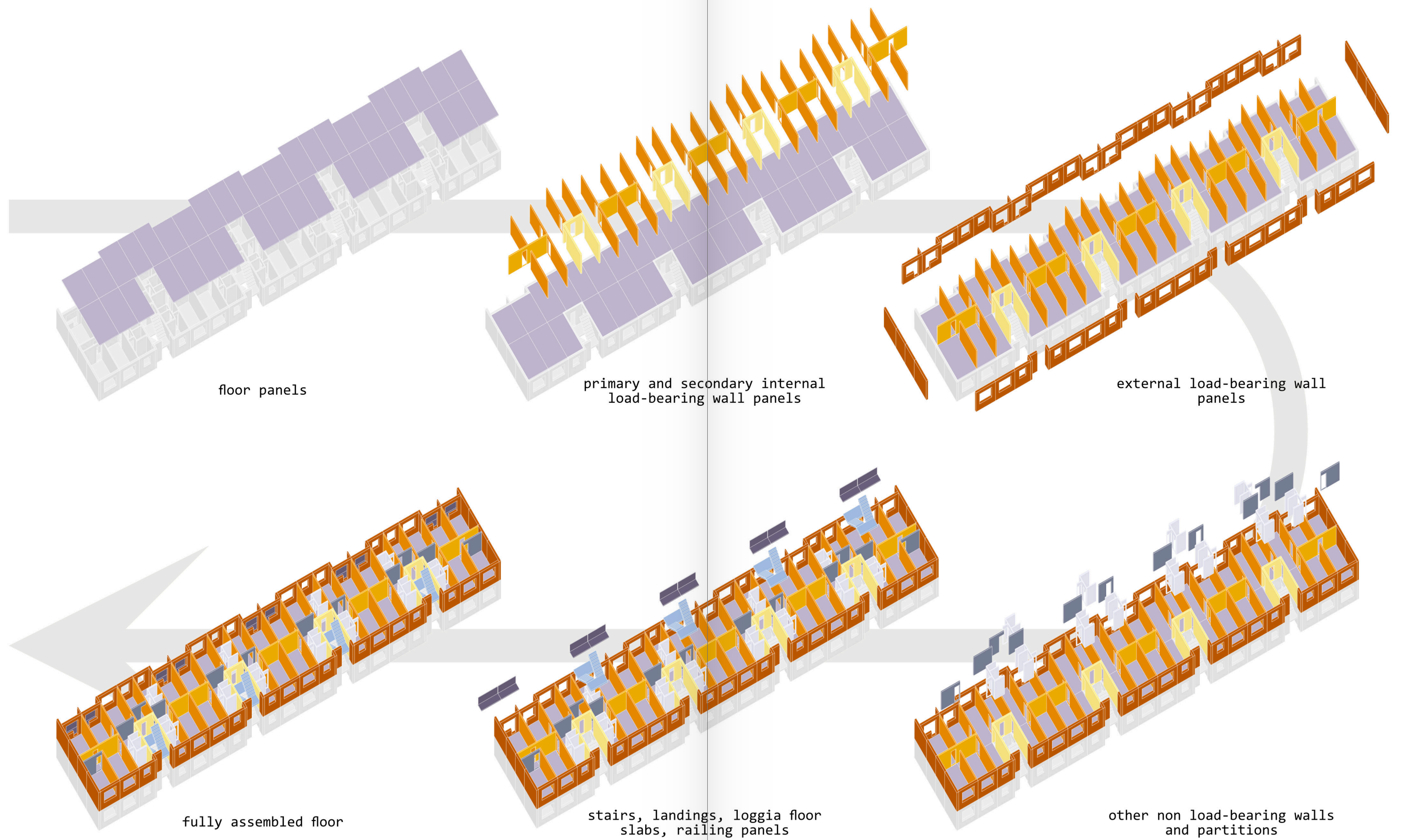


Figure 28. Original building assembly sequence

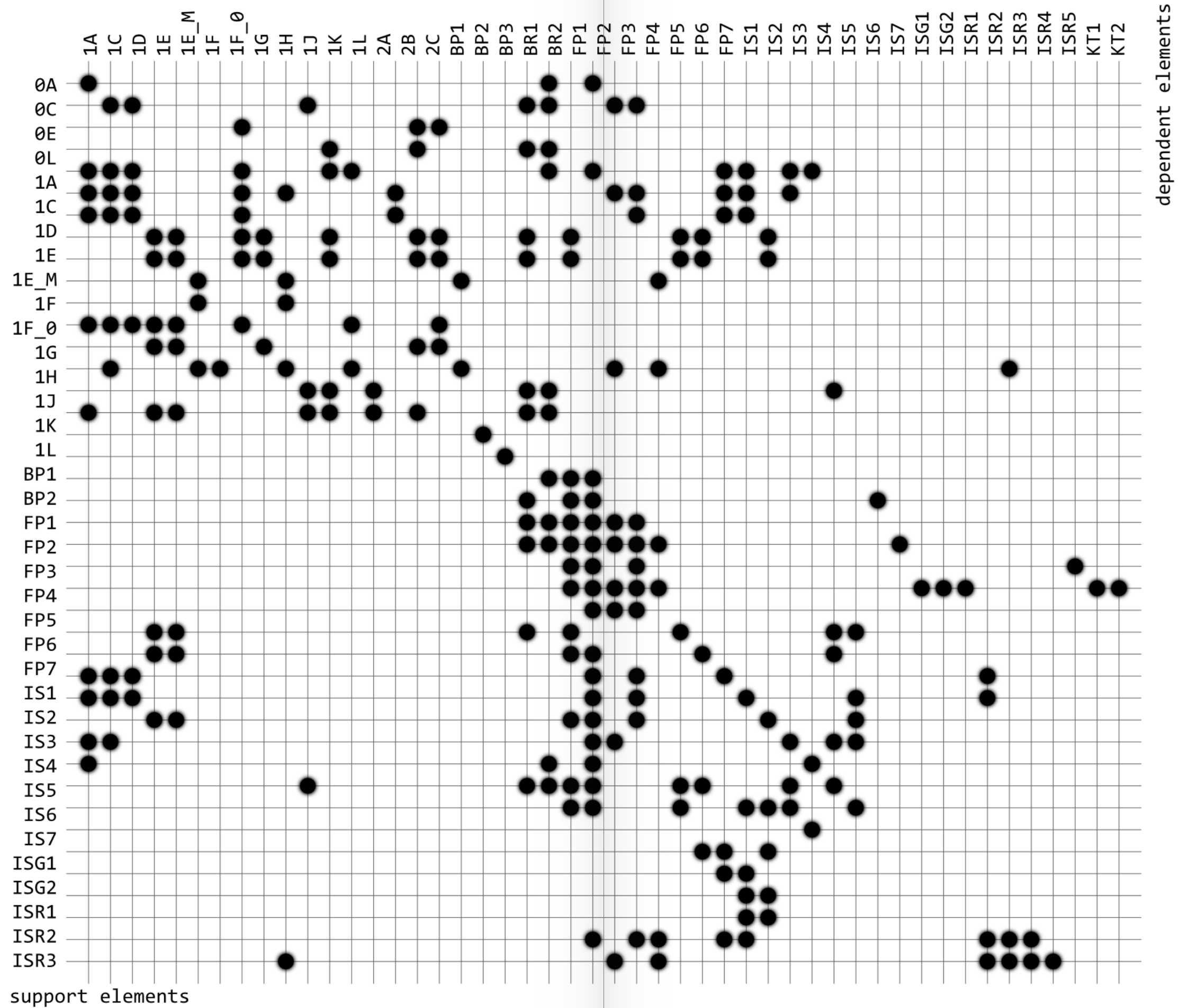


Figure 29. Extracted dependency matrix mapping structural relationships

5.3 TOP-DOWN VERSUS BOTTOM-UP LOGIC

Initially, the anticipated process was expected to be quite linear: based on the dependency matrix, specific structural rulesets would be defined. For example, if a transverse load-bearing wall were to be targeted, four absolute scenarios for what must happen would be formulated, ranging from safe removal to mandatory insertion of a new structural beam. However, when attempting to create an algorithm to automatically apply these rules, an issue was encountered: the raw dependency matrix did not hold the actual data of a specific wall supporting a specific element. In order to achieve that, a significantly more complex dependency mapping script would need to be created, where if a wall is targeted for removal, all dependent elements would automatically be flagged as well. This realisation prompted a re-evaluation of the planned workflow and an exploration of three different methodological paths that could potentially be taken to alter this building.

Option 1

This option revolves around the previously discussed scenario definition. Each support element of the building would have four different scenarios assigned, and using a grayscale image sampler, these scenarios would be applied, taking into account all dependent elements connected to these

supports. However, since this approach would be strictly physics-based, it would present several critical issues.

The structural relations of this building work in a reverse-triangle load distribution. If an element were to be removed on the ground floor, it would trigger a catastrophic cascade, risking the collapse of the entire building. Essentially, all alterations on the ground floor would affect the alterations on all floors above, meaning that the entire design outcome would depend on the level to which the intervention image sampler is applied.

That brings out the issue of architectural need versus structural reality: in the urban analysis (see Section 2.2.4), it was identified that the greatest urban deficiency of the neighbourhood is a lack of public, especially community and gathering, spaces. Logically, it would seem fit to propose these on the street level – the ground floor – for easier accessibility and the ability to separate them from the primary residential function. However, in this approach, the ground floor would be the least modifiable – if an element were to be removed here, the whole building would risk collapse, or extensive structural supports would need to be inserted. The most alterable level would instead be the top floor, which is the least suitable for this particular functional injection. Additionally, this option would involve an excessively complex dependency script creation.

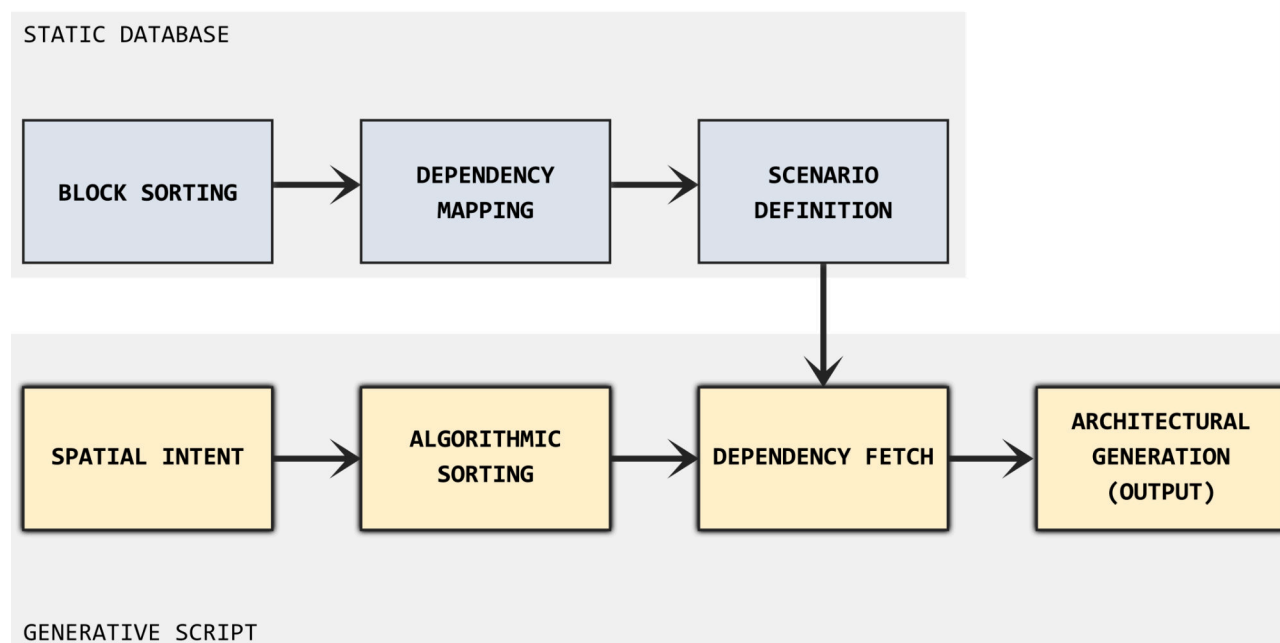


Figure 30. A computational workflow diagram for Option 1

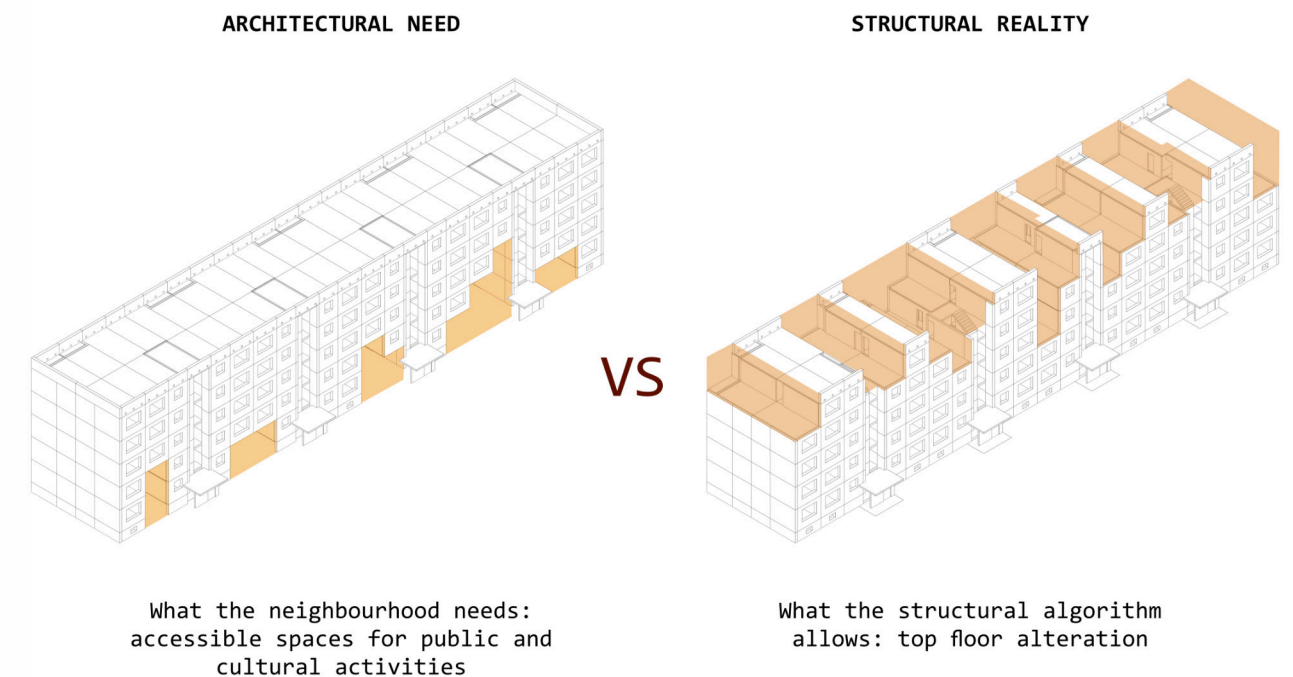


Figure 31. Comparative diagram illustrating architectural need versus structural reality in top-down floor plan alterations

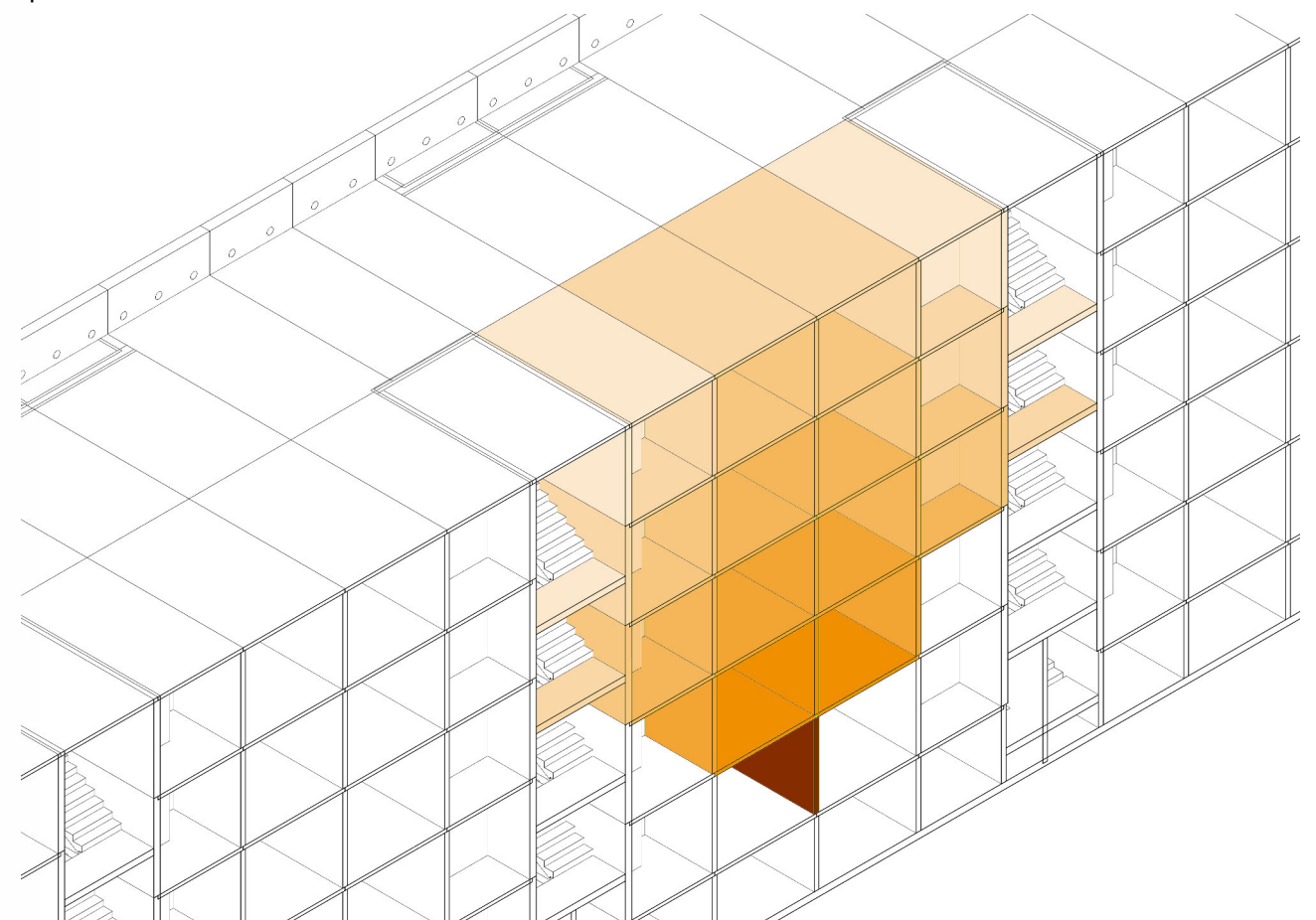


Figure 32. Reverse-triangle load distribution principle in a building section

Option 2

For the second option, in contrast to the first, the dependency script would be bypassed altogether, and the building would be treated as horizontal slices of the five floor plans. It would also involve the scenario definition, but in this case, the scenarios would not depend on structural relations, but would rather act as a simple decision-making pipeline between four states: remove, replace, partial-replace, or preserve as-is. The script would also utilise a grayscale image sampler, but in this case, to carve out each floor plan separately. This method would present incredible spatial freedom on the floor plan to inject new functions.

However, ignoring physics brings up an obvious dilemma: it risks catastrophic collapse in the vertical dimension. If the script were to remove a wall on the ground floor but keep the wall directly above it, a floating wall would remain. Fixing this would require extensive manual overrides to check each element and its dependencies. Additionally, this layered two-dimensional approach holds the possibility of ending up with a very fragmented design.

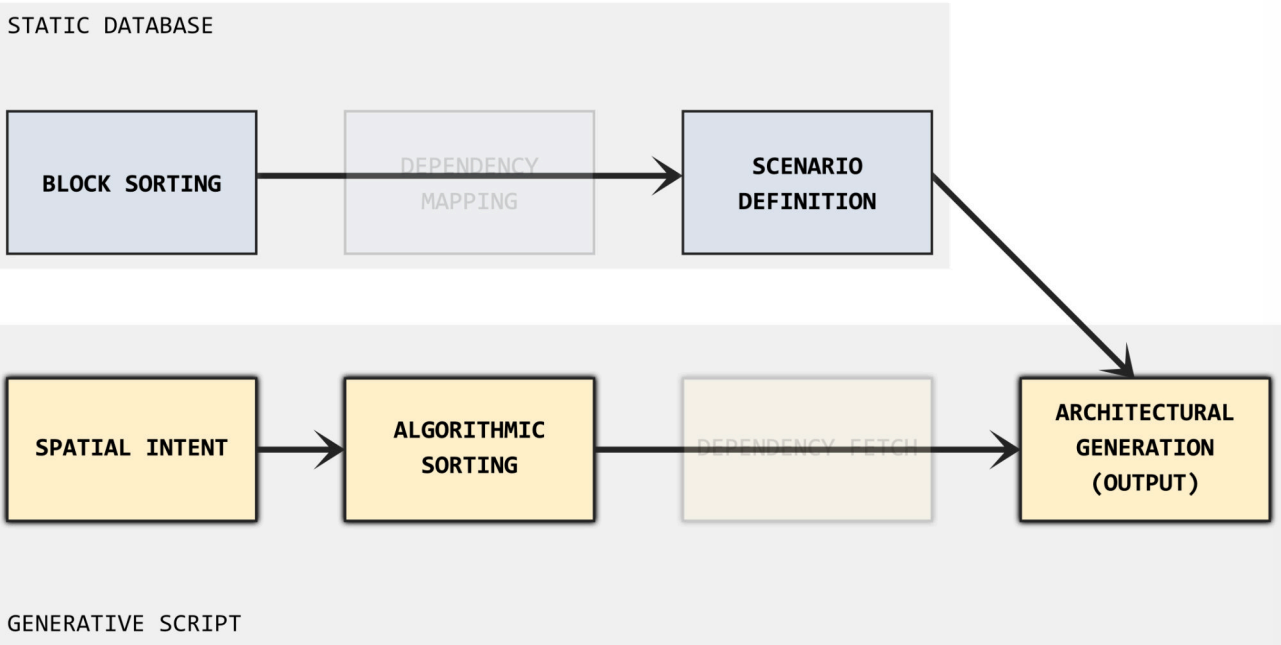


Figure 33. A computational workflow diagram for Option 2

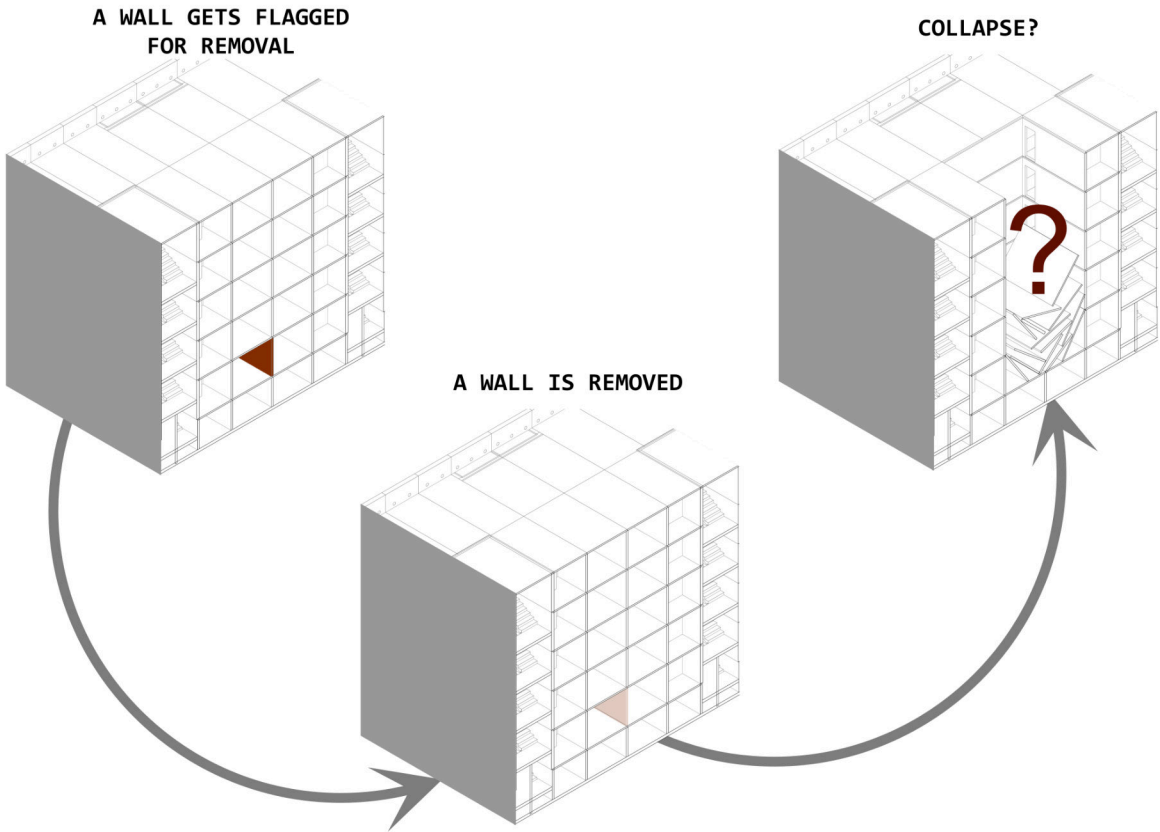


Figure 34. An illustration showing possible structural collapse consequences if the dependency script is bypassed

Option 3

Option 3 takes a step back, and, instead of looking at the building at the element scale, it looks at it at the module scale. This workflow would also bypass the dependency mapping, but since it would operate on a modular, not element, scale, the manual dependency checking and sorting would be less extensive.

In this option, a grayscale image sampler would again be applied to flag the modules for either removal or replacement. Then they would be sorted according to the vertical situation – if a module is on the top floor, it could be safely removed. If it is a middle module, but the module above it is also removed, then it could be removed safely as well. If it is a middle module and the top modules are not removed, a new support structure would need to be inserted instead.

The architectural possibilities for this approach are inherent, and since it would work at a modular scale instead of an element scale, it could create a less fragmented design. However, there are several issues with this approach as well: removing whole modules would expose the current interior boundaries to the outside climate, raising the question of how to mitigate that. Another challenge, since the thesis revolves around reorganisation, not only removal, is the return of these modules. Either they would be returned to the building in their current shape and boundaries, which is highly experimental

and would require extensive external structures to support the extra load, or these removed modules would need to be dismantled and returned back to the building in a more informal arrangement.

Following the analysis of all three options, a decision was made to proceed with Option 1: physics-based element extraction. This approach is tied closely to real-world conditions and integrates the micro-scale relationships between precast concrete elements, which is the fundamental mechanism ensuring structural stability within these assemblies. Additionally, as discussed in Section 4.2, thoroughly calculating these dependencies is ultimately the only way to prevent a progressive collapse of the building. Therefore, despite the computational complexity required to build an advanced script for dependency identification, it was deemed the only structurally and architecturally responsible path forward for this project.

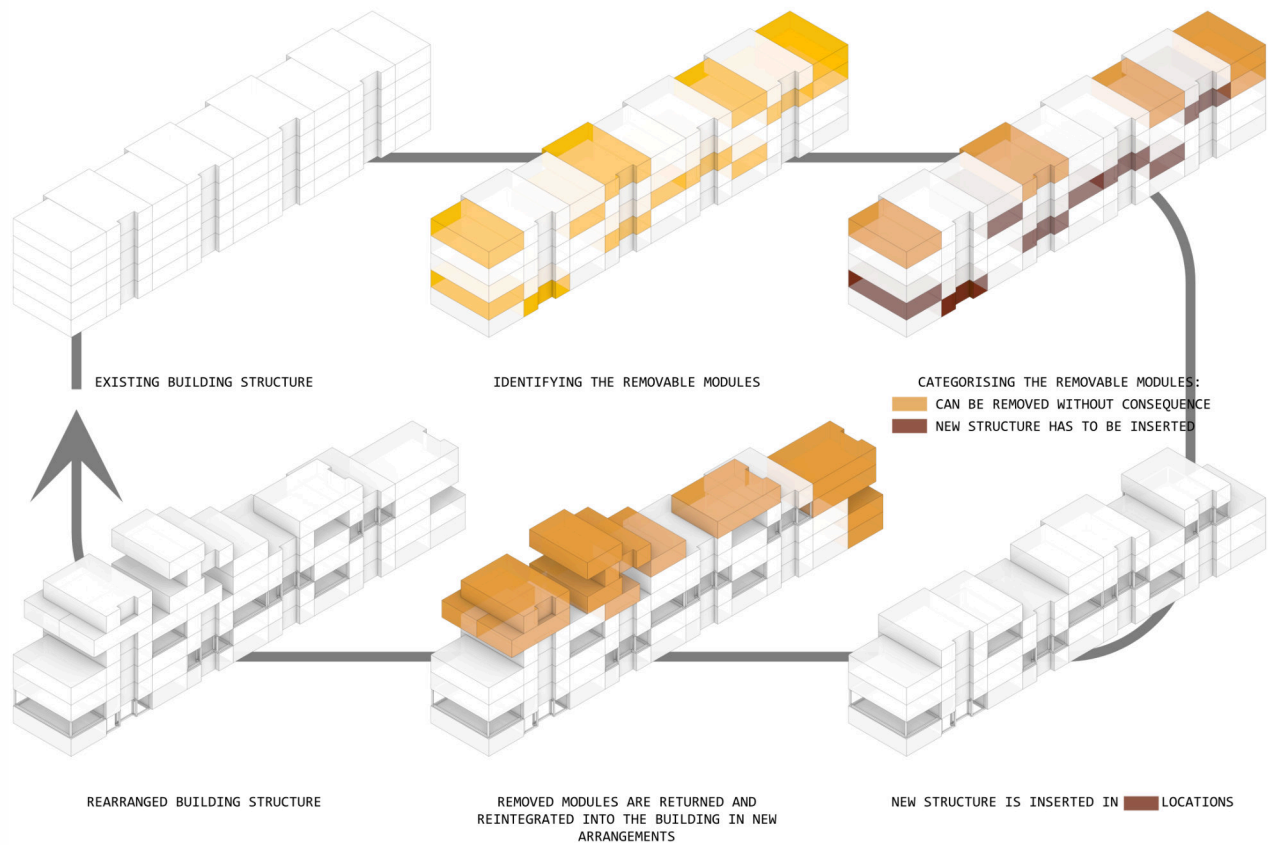


Figure 36. An illustration of the possible alteration process if Option 3 is developed further

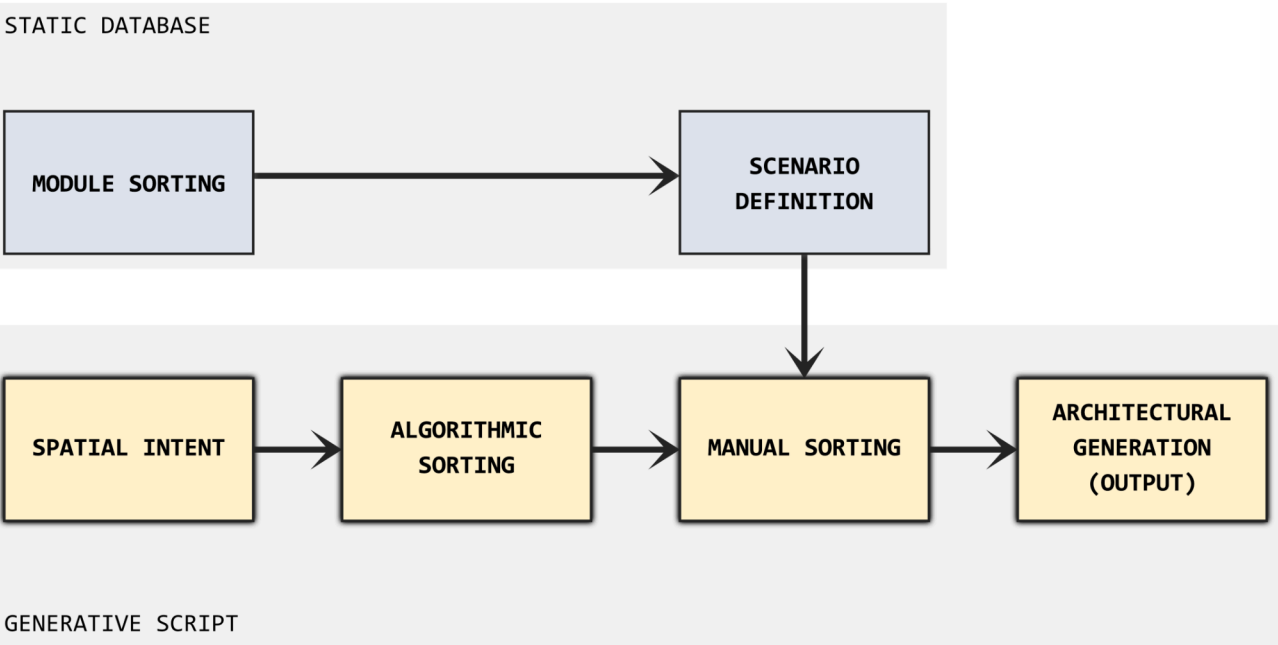


Figure 35. A computational workflow diagram for Option 3

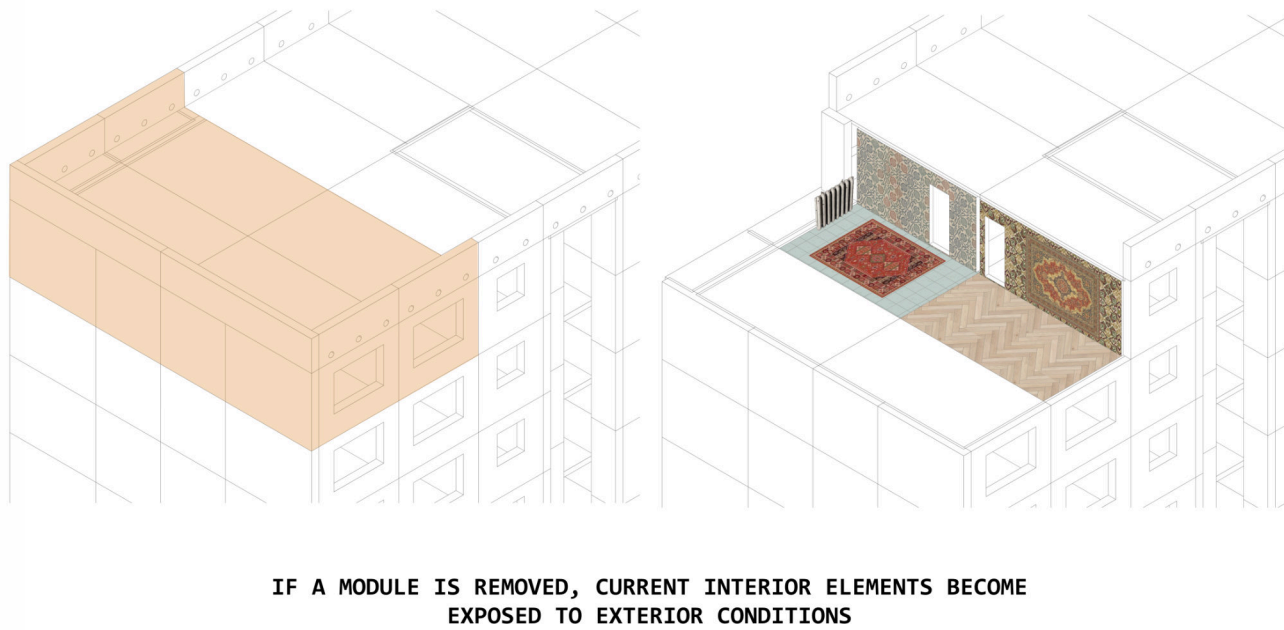


Figure 37. An illustration showing the spatial concerns of whole module removal

6. SYSTEMIC ADAPTATION

This chapter details the creation of the computational system, which functions as the “brain” of this thesis and the primary generative tool. It establishes the rulesets created and justifies the architectural decisions made to facilitate safe partial disassembly and reorganisation. It evaluates the relevance of the dependency matrix and describes the custom Grasshopper scripts in detail, explaining the logic behind the decision-making process. It concludes with a comprehensive explanation of how these sub-routines are synthesised into a recursive, user-guided generative tool.

6.1 THE DEPENDENCY WEB

Following the decision to pursue the physics-based option requiring complex dependency mapping, the dependency script had to be created. The unique element relationships and hierarchies were already defined as discussed in Section 5.2, thus, a framework to use this raw CSV data in Grasshopper to tie these relationships back to the physical geometries had to be established. In the raw data, the unique element identifiers were used to describe the relationships, and since the digital twin was created with the corresponding unique identifiers, the string (Str) data matching could be utilised.

This approach requires two distinct data streams for the geometries: one for the targeted support element (a specific referenced block geometry) and one for the potential dependents (referenced all block geometries in the Rhino file). Next, the block references in both streams are filtered by their identifiers to be able to control the computational process downstream. The data in both streams is then merged into two lists, from which the block references are exploded and bounding boxes are created. This step is necessary both because block references are not native to Grasshopper and to simplify the dependency checking process. Parallel to this, the block names are extracted to be cross-referenced against the CSV data. Concurrently, the raw CSV data columns are deconstructed into separate, synchronised data streams – support element, dependent element, relationship type, and structural hierarchy.

The next step involves the creation of the “brain” of the dependency script, which was executed via a Python script within the Grasshopper environment. The inputs for this script are the geometry-based data from both streams previously discussed (original blocks, bounding boxes, and block names), the data from the CSV (support element, dependent element, relationship type, and structural hierarchy), as well as locally defined values such as a filter for the hierarchy level, a filter for the relationship type and a geometric tolerance. When an element is targeted for removal (the support element), the algorithm initiates a spatial “radar”, isolating the surrounding elements within the specified tolerance. It mathematically evaluates the Z-axis coordinates to categorise the physical

intersection as either a vertical stack or a lateral bracing. Crucially, the algorithm then validates this physical geometry collision against the CSV dictionary to ensure that the geometric collision matches the relationship described in the data matrix. Only if both the geometric orientation and the described relationship data match, is the element flagged as dependent.

Inherently, this script itself only identifies direct dependencies, i.e., it only considers elements that are physically touching within the tolerance limits. Therefore, to accurately depict the reverse-triangle load distribution discussed in Section 5.2, an Anemone loop was introduced to operate the script recursively and accurately simulate the cascading nature of progressive collapse. The script therefore logs the flagged elements into a “next support element” container and runs the script recursively a predefined number of times, or until all the elements in the building are flagged.

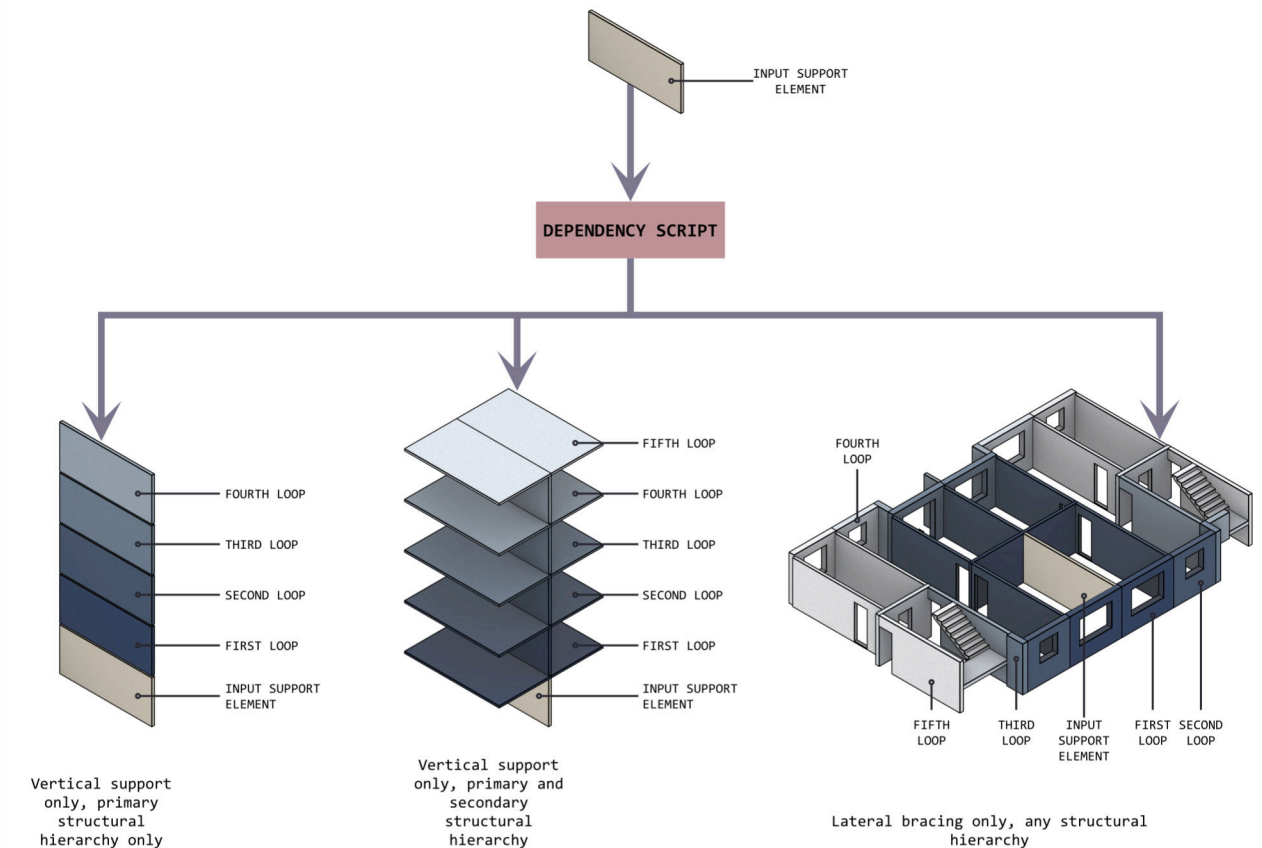


Figure 38. A scheme showing the principle of the dependency script with varied structural hierarchy and element relationship type inputs

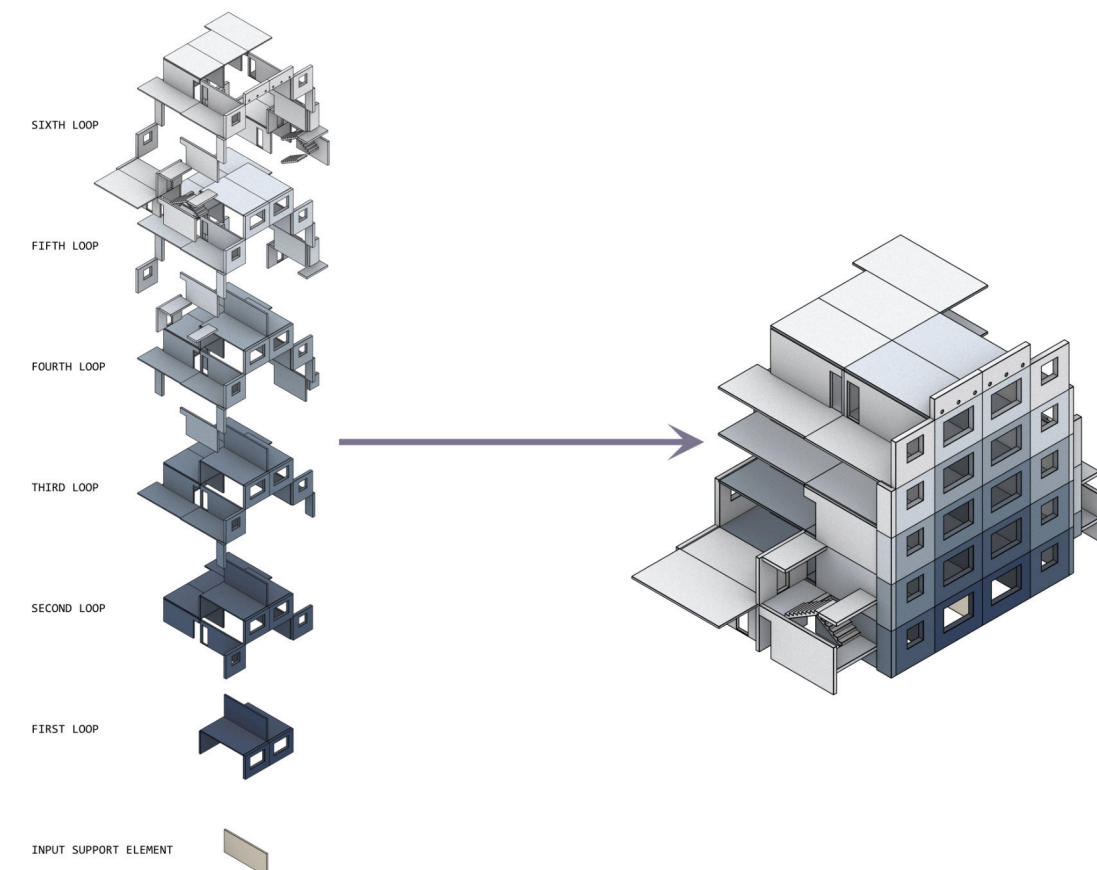


Figure 39. A scheme showing the principle of the dependency script with all structural hierarchies and element relationship type inputs enabled

6.2 ELEMENT-LEVEL TRANSFORMATION SCENARIOS

Having established the dependency script, the scenarios of what happens to the elements had to be defined, and scripts for these transformations had to be developed. The chosen scenarios can be categorised into three primary groups: removal, partial removal, and replacement, however, not all scenarios are applicable to every element in the typology. The applicability of these scenarios is shown in Table 6. These scripts were first developed as modular sub-routines, functioning independently from the dependency script.

As illustrated in the table, the decision-making process for these applications was rooted in the physics-based logic. Non-load-bearing elements can be removed without consequence, therefore there is no justification to alter these elements with partial removal or replacement scripts. Partial removal is aimed at load-bearing walls in order to preserve the primary load bearing nodes. This approach allows for either the expansion of existing openings (in the case of

external walls) or the creation a new opening in the middle of the wall (in the case of interior walls), while preserving the top margin for floor slab support and side margins for lateral bracing. For floor slabs, the structural logic dictates that they can either be fully removed (depending on the support element) or an opening can be cut into them, preserving the margin that ensures the stability of the walls resting above. Based on this scenario definition, individual, task-specific Grasshopper scripts were created to execute these transformations.

Element type	Element identifier	Removal	Partial removal	Replacement
Sub-surface foundation panels	0x	☐	☐	☒
Exterior wall panels	1x	☐	☒	☒
Parapet wall panels	2x	☒	☐	☐
Transverse interior load-bearing wall panels	ISx	☐	☒	☒
Longitudinal interior load-bearing wall panels	ISGx	☐	☒	☒
Staircase wall panels	KTx	☐	☒	☒
Non-load-bearing wall panels	ISRx	☒	☐	☐
Floor slab panels	FPx	☒	☒	☐
Loggia slab panels	BPx	☒	☐	☐
Loggia railing panels	BRx	☒	☐	☐
Bathroom blocks	WCx	☒	☐	☐
Stair run blocks	STx	☒	☐	☐
Stair landing panels	SLx	☒	☐	☐

Table 6. Applicability of created scenarios for each element type

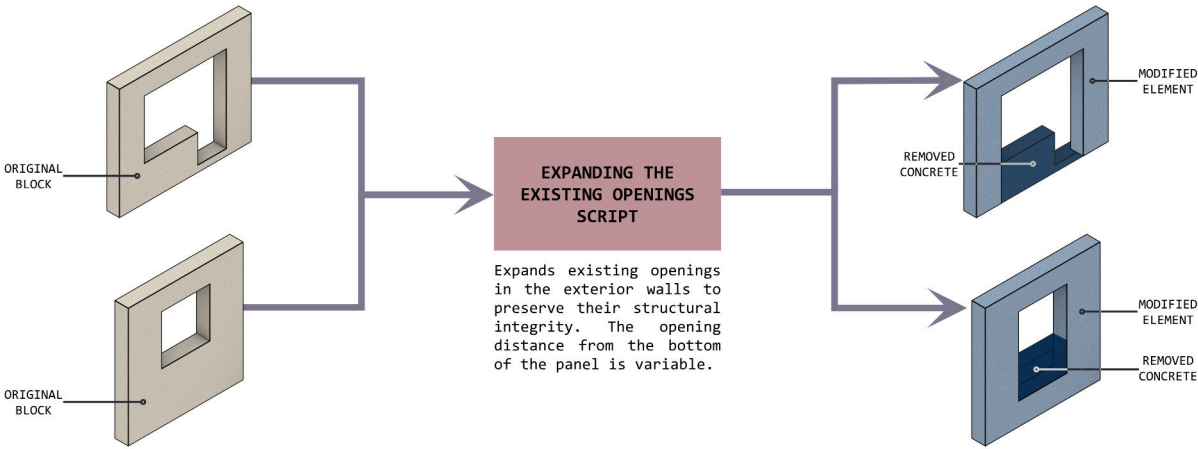


Figure 40. The principle of the partial removal script for exterior wall panels

Partial removal

External walls

To preserve the structural integrity and load-bearing capacity of the exterior walls, a decision was made to expand the preexisting openings in the vertical dimension. Because the structural lintel above the window opening distributes the overhead load to the sides of the opening, the section of the wall below the sill is not load-bearing and therefore can be safely removed. Consequently, this approach subjects only the exterior wall panels with pre-existing openings for the application of the script.

The input for this script consists of the original geometries of the exterior wall panels (block references), which are then sorted into two streams: panels with openings and panels without. The panels without openings are discarded from the process, while the blocks with openings are exploded to extract their native boundary representations (Brep). These Breps are then passed to a custom Python script that analyses the existing openings, extracts their coordinates, and

expands these voids as separate Breps to act as “cutting boxes”. Another input for this script is a bottom offset – a variable defining the distance from the bottom of the panel to the base of the new opening, which allows for improved control of the architectural process. Finally, a solid difference operation is performed between the original geometry Brep and the new “cutting box”, producing the adapted geometry. Additionally, a solid intersection is performed on these two sets of Breps to isolate the leftover concrete volumes, which are quantified for further material reuse.

Partial removal

Interior walls

In contrast to exterior walls, the objective for partial removal in interior wall panels is not to expand existing openings, but rather to create a new opening in the centre of the panel, retaining a top margin for floor support and side margins for lateral bracing. Therefore, the algorithm for interior walls is somewhat more complex.

The first step is to sort the panel geometries according to the position of their original openings: panels with no openings, edge openings, near-edge openings, and central openings. This is necessary to prevent the generation of anomalous geometries, such as overlapping voids and spatially impractical micro-segments of leftover wall. These blocks across the four data streams are then exploded to extract the native geometry Breps. The core workflow for all streams is to extract the geometric centroid of the panel and create a new “cutting box” Brep at these coordinates. The dimensions of these cutting boxes are controlled via a

numeric side margin and top margin inputs. To simplify this workflow, the geometries for panels with pre-existing corner openings were modified manually in Rhino, referenced back into Grasshopper, and oriented to replace the original geometries. As with the exterior walls, the residual concrete masses were simultaneously extracted.

Floor slabs

The workflow for new void creation in the floor slab panels follows an analogous logic to the interior walls, facilitated by the simpler geometries of the slabs with no pre-existing openings. A margin input was again introduced to control the dimensions of panel retained for structural support alongside the perimeter. The residual concrete was similarly extracted for downstream material reclamation.

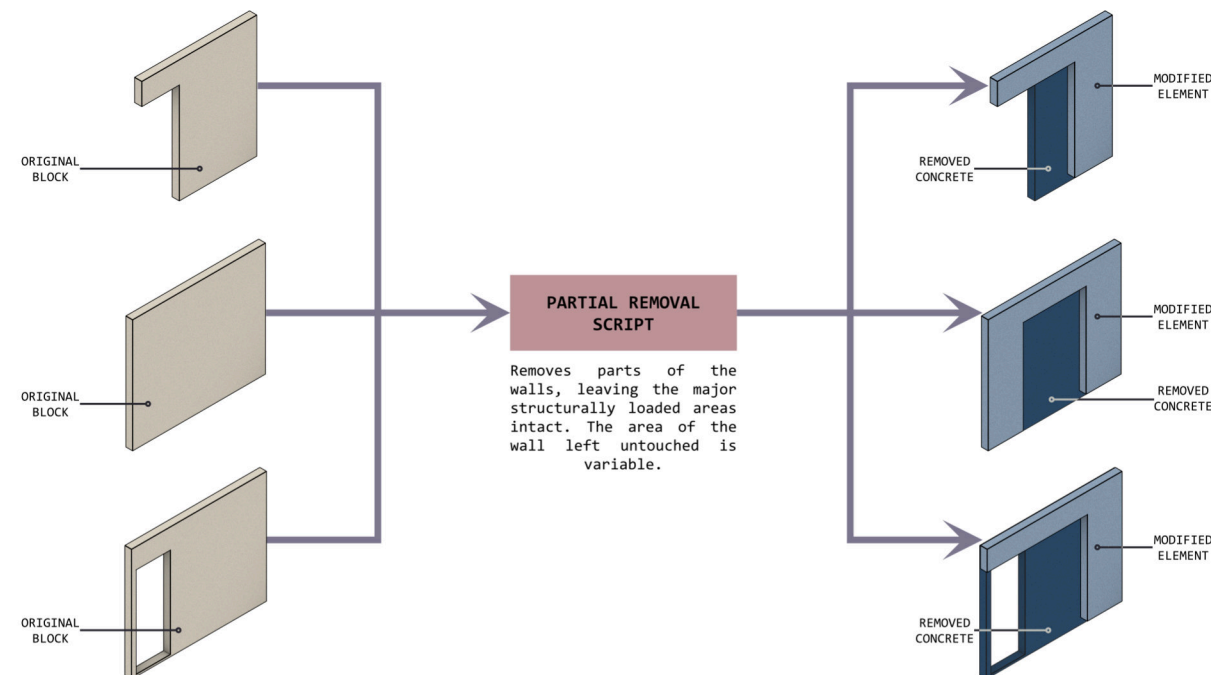


Figure 41. The principle of the partial removal script for interior wall panels

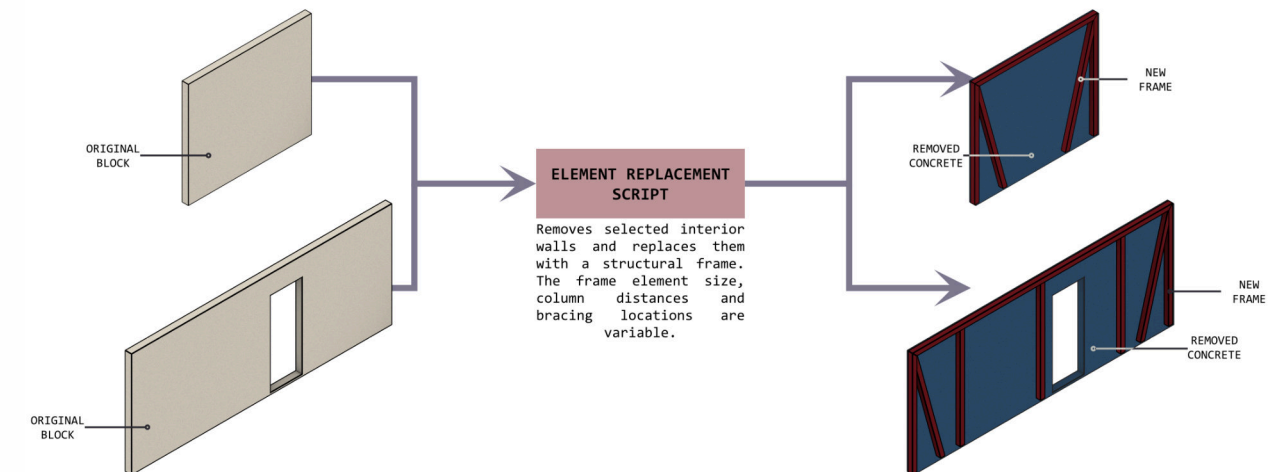


Figure 42. The principle of the replacement script for interior wall panels

Replacement

Interior walls

The fundamental logic behind the replacement script is straightforward: if a load-bearing wall is removed, the loads must be redistributed; therefore, it must be replaced by a column-beam frame. This is achieved by firstly filtering the interior walls into two categories - transverse and longitudinal - to ensure correct frame orientation. Next, the native Brep geometries are extracted, and bounding boxes are created, because, in this scenario, it does not matter whether the wall panel has an opening or not. Then the centre plane along the longitudinal axis of the wall is extracted to determine the insertion plane for the new structural frame. The frame is created as a simple extrusion along these edges with variable beam/column size controlled through a numeric input. This constitutes the primary structural frame.

Since the dimensions of these wall panels are not uniform, and some span beyond the optimal length for a load-bearing beam, a

parallel script computing the length of the original wall is utilised to insert additional columns if the wall length exceeds a predefined numeric threshold. Additionally, since these load-bearing interior walls provide lateral bracing for both one another and the exterior wall panels, diagonal bracing is introduced at the ends of these new-defined frames to ensure structural stability. The desired distance of the bracing from the wall corner is dictated via another numeric input.

Replacement

Exterior walls

The script for replacing exterior wall panels with frames follows a logic analogous to the interior walls, except that no intermediate column grid is introduced, as the exterior wall panels are smaller in dimension, and the diagonal bracing spans across the entire new frame. However, as discussed in Section 4.2, the loads in the exterior walls of this building operate as a macro-masonry assembly, where each element supports all surrounding elements. To prevent progressive collapse in the event of external wall removal, an exoskeleton externally bracing the newly defined frame must be introduced. This exoskeleton is defined via an input from Rhino – a reference grid curves for the proposed exoskeleton to follow. It consists of beams and columns, which are then mapped against the coordinates of the removed elements and new frames, and subsequently generated only where needed.

6.3 ASSEMBLY-LEVEL TRANSFORMATION MATRIX

With the modular sub-routines established, the next step was to assign them to overarching operational scenarios. Essentially, the subsequent task was to categorise each element and its transformation potential into a logical decision-making matrix. The objective of this matrix was to take these element-level interventions and categorise them into assembly-level scenarios, scaled by their level of intervention, execution complexity, and the provisional cost of extraction. Therefore, four distinct scenarios were created, as shown in Table 7.

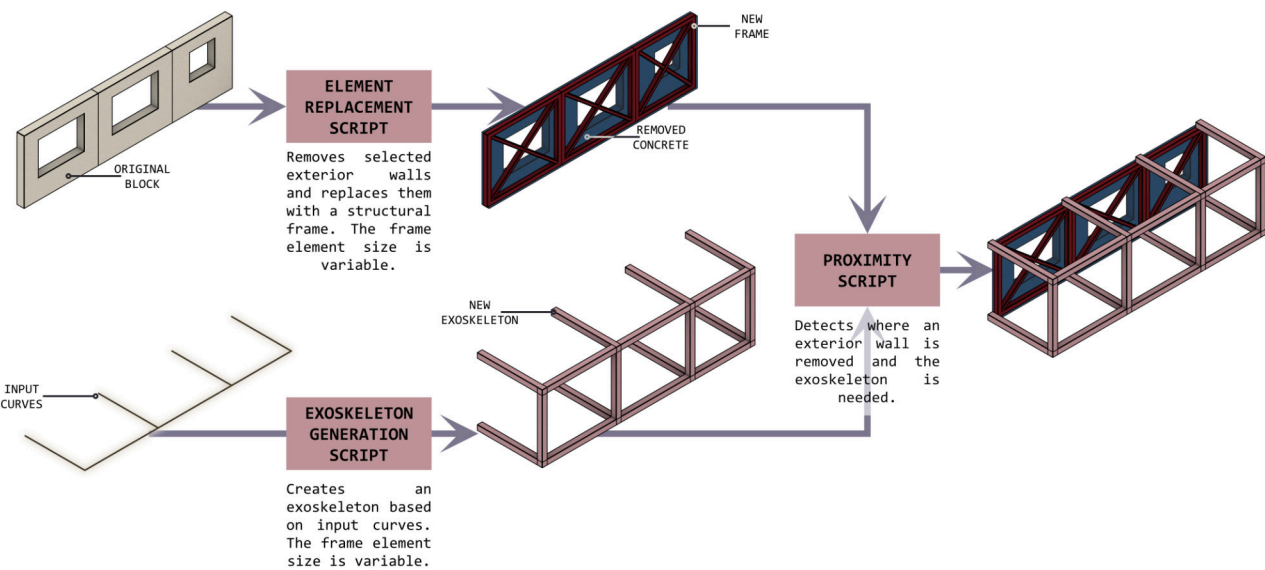


Figure 43. The principle of the replacement script for exterior wall panels

Element type	Element identifier	SCENARIO A	SCENARIO B	SCENARIO C	SCENARIO D
Sub-surface foundation panels	0x	-	-	-	replacement
Exterior wall panels	1x	-	partial removal	-	replacement
				partial removal	
Parapet wall panels	2x	-	-	-	removal
Transverse interior load-bearing wall panels	ISx	-	partial removal	replacement	-
					partial removal
					replacement
Longitudinal interior load-bearing wall panels	ISGx	-	partial removal	replacement	-
					partial removal
					replacement
Staircase wall panels	KTx	-	partial removal	replacement	-
					partial removal
					replacement
Non-load-bearing wall panels	ISRx	removal	removal	removal	removal
Floor slab panels	FPx	-	-	-	-
		partial removal	partial removal	partial removal	partial removal
Loggia slab panels	BPx	-	-	-	removal
Loggia railing panels	BRx	-	-	-	removal
Bathroom blocks	WCx	removal	removal	removal	removal
Stair run blocks	STx	-	-	-	-
Stair landing panels	SLx	-	-	-	-

Table 7. Scenario assignment to each element type

Scenario A only operates on a fundamental level of removal concerning only non-load bearing walls, therefore this scenario can be categorised as a very low-intervention approach, as it does not alter the load-bearing elements at all. Additionally, partial removal can be applied to floor slabs if desired. It is assumed that the execution of this scenario would be relatively low-cost and structurally non-disruptive.

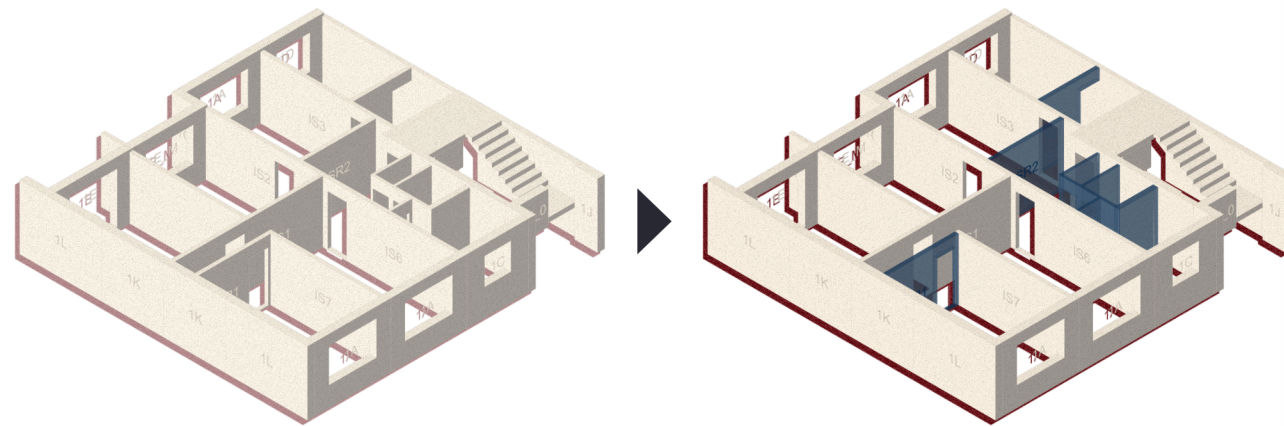


Figure 44. Scenario A application example on a partial building floor

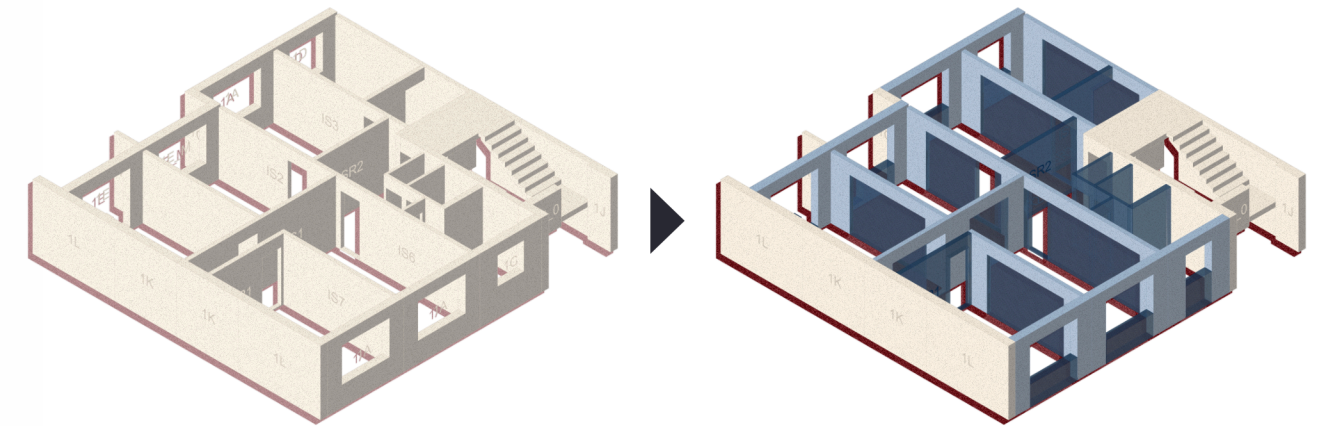


Figure 45. Scenario B application example on a partial building floor

Scenario B represents a higher level of intervention, targeting non-load-bearing elements for removal, but also targeting load-bearing elements for partial removal. Since the partial removal (as discussed on the Section 6.2) aims to preserve the load-bearing capacity of the elements, it is assumed to not trigger the dependency script for the dependent elements. Therefore, this scenario can be categorised as a medium intervention, because the partial subtraction of concrete mass is labour-intensive, but the overall structural scheme is not compromised. It is assumed that the execution of this scenario would carry a medium cost, as it requires extensive precision sawing and milling of the elements but does not require new structural additions such as steel frames.

Scenario C introduces a high level of intervention, as it targets primary load-bearing elements – specifically interior walls – for removal and replacement. Extracting these elements from their original positions is assumed to be highly labour-intensive, and the insertion of new steel frames requires the addition of raw materials. Since this scenario targets load-bearing elements for removal, it triggers the dependency script, computationally dictating the fate of the dependent elements. This scenario is assumed to be high cost, as it requires extensive cutting, milling, and the insertion of new steel frames.

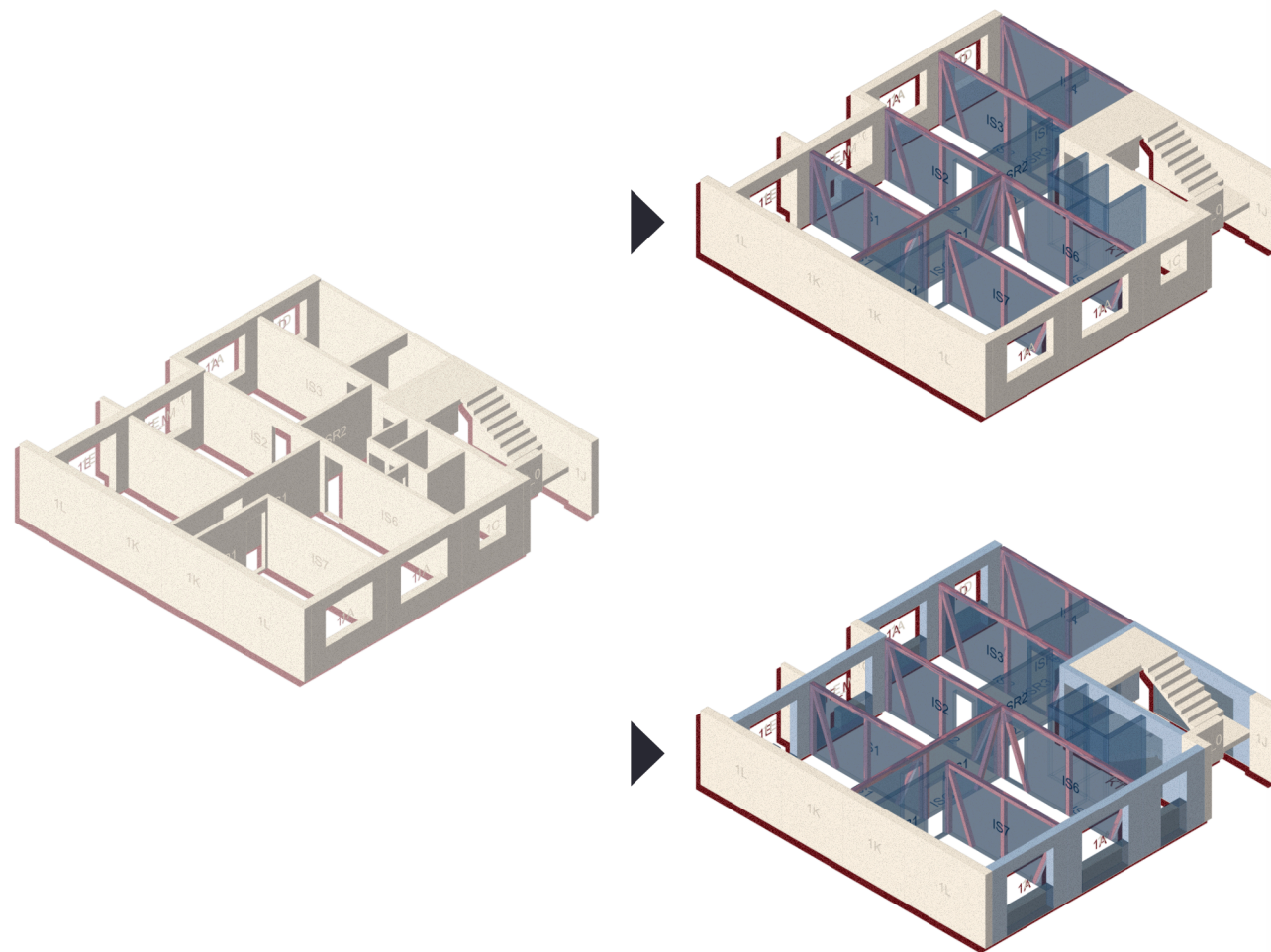


Figure 46. Scenario C application example on a partial building floor

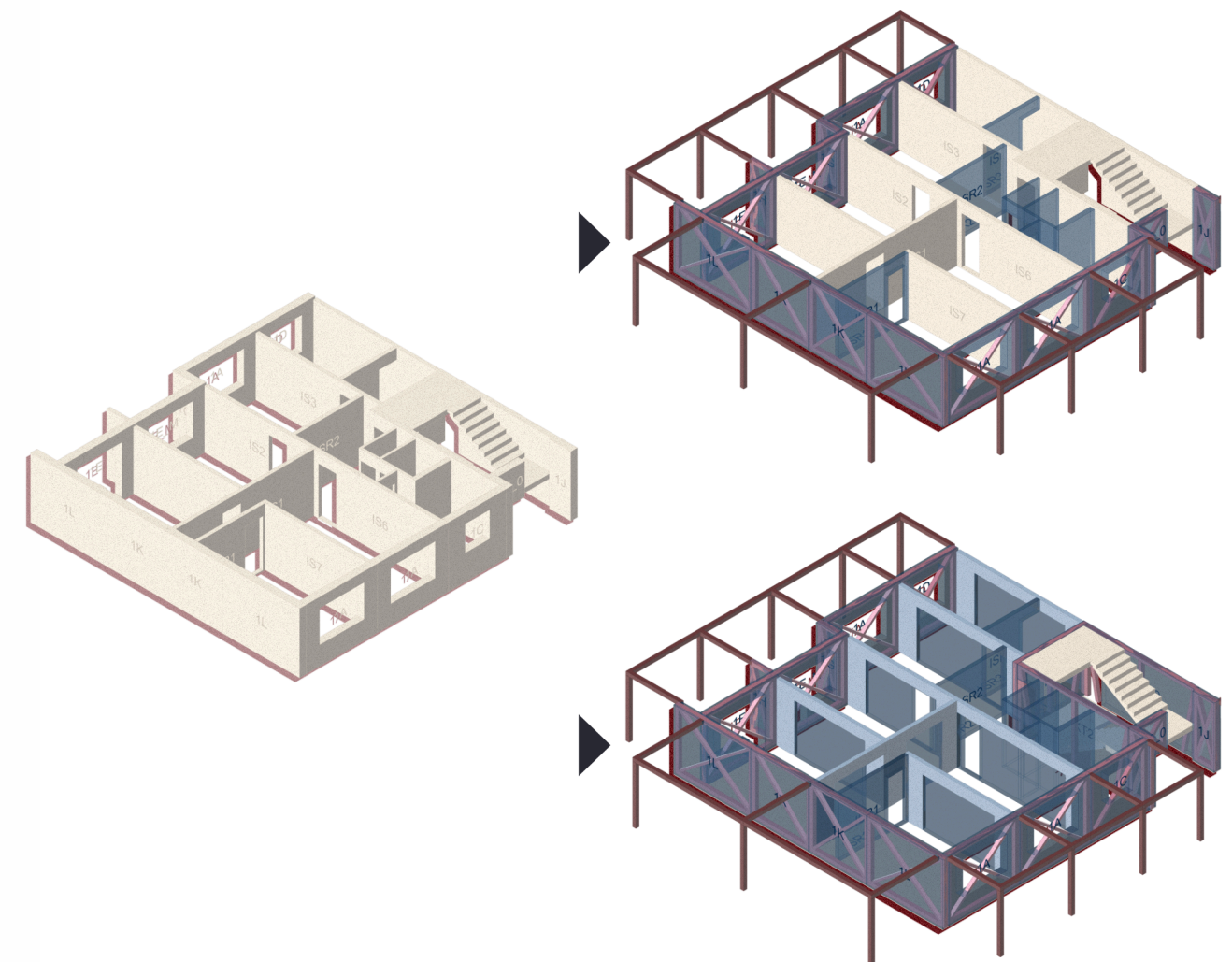


Figure 47. Scenario D application example on a partial building floor

Scenario D most radical level of intervention. It targets the load-bearing elements – specifically exterior walls – for removal and replacement. Extracting these elements is labour-intensive and the newly added frames are quite large in dimension in order to support the cascading loads. This scenario also triggers the dependency script. Additionally, in this scenario, an exoskeleton is added to support the exterior wall frames, significantly increasing the amount of raw material needed and categorising this scenario as highly resource-intensive and expensive.

It should be noted that these scenarios operate cumulatively: higher levels of intervention, such as scenario C and D, inherently encompass the scenarios of lower levels, which can be observed in the Table 8 as several possible scenarios for an element. The decision making is embedded in the script as a manual selection tool, and the selection is done depending on the users desired outcome. This therefore highlights the need for a variable versus locked in ruleset matrix creation.

6.4 LOCKED VERSUS VARIABLE RULESETS

As established in the scenario matrix, certain elements are assigned a “locked” or static scenario of alteration, while other elements have several possible outcomes and are therefore categorised as “variable”. This calls for an updated ruleset matrix to clearly define these computational rules. This updated matrix is shown in Table 8. Essentially, this matrix refines the information shown in Table 7 from the previous section, but it is organised to provide a clearer overview of which elements are variable, facilitating their integration into the script.

As can be observed in the table, specific elements always remain unmodified (○) across all scenarios, such as stair runs (STx) and landings (SLx). This restriction ensures the critical vertical connections and fire exit routes are preserved regardless of the intervention scale.

Conversely, certain elements are locked as permanently “modified” (●). For example, non-load-bearing partitions (ISRx) and sanitary blocks (WCx) are automatically targeted for removal in all scenarios. This reflects the spatial obsolescence identified in Section 4.4, dictating that in order to adapt the layouts to modern conditions, these strict spatial boundaries must be cleared.

Finally, the elements classified as “variable” (●) represent the generative flexibility of the algorithm. For these components the script does not force an absolute global action; instead, the decision-making is delegated to the user of the tool, represented as a manual decision-making “node” within the Grasshopper script. These variable elements represent the cumulative operation of the scenarios: as can be observed in Table 8, the higher the level of intervention, the greater the count of variable elements becomes.

Element identifier	SCENARIO A	SCENARIO B	SCENARIO C	SCENARIO D
0x	○	○	○	●
1x	○	●	●	●
2x	○	○	○	●
ISx	○	●	●	●
ISGx	○	●	●	●
KTx	○	○	●	●
ISRx	●	●	●	●
FPx	●	●	●	●
BPx	○	○	○	●
BRx	○	○	○	●
WCx	●	●	●	●
STx	○	○	○	○
SLx	○	○	○	○

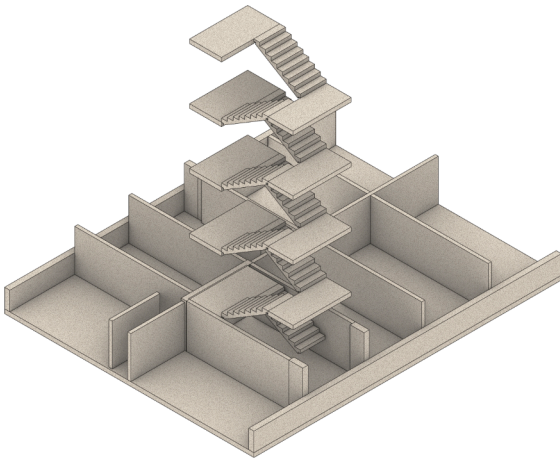
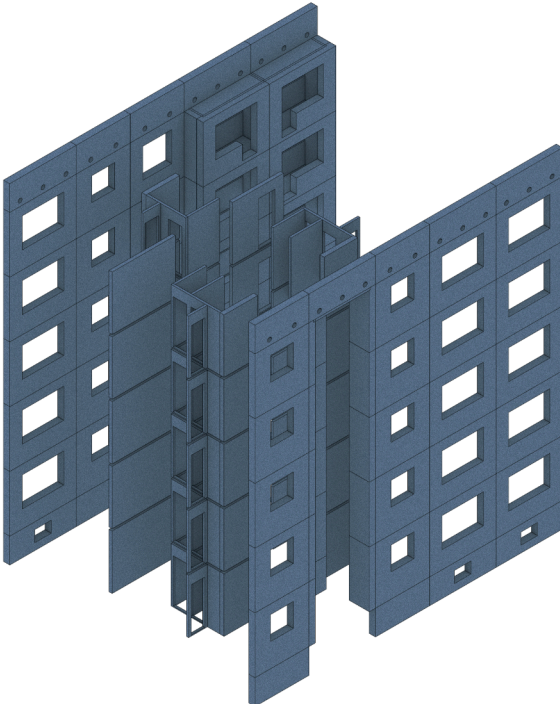
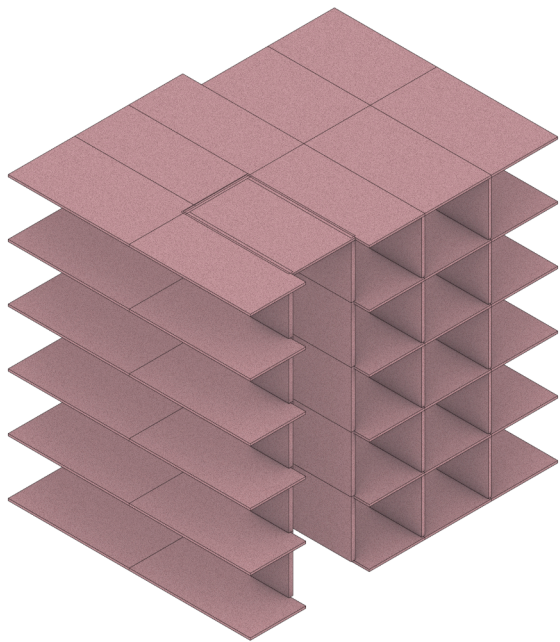
Table 8. A matrix defining unmodified, variable and locked state for each element type within the scenarios

ELEMENTS WITH VARIABLE SCENARIOS

ELEMENTS WITH LOCKED IN SCENARIOS

ELEMENTS LOCKED AS UNMODIFIED

Figure 48. Elements for Scenario D with defined unmodified, locked and variable states



6.5 GRAYSCALE SAMPLER AS A TOOL

Although the application of variable scenarios is defined as a manual input, there was still one large question remaining – how are these global scenarios applied spatially across the building floor? The method of generating pseudo-random numbers as inputs via Python or native Grasshopper was discarded early in the process, as it offers little to no control over the output. Instead, because there are only four distinct scenarios, the Grasshopper image sampler was identified as an optimal tool to facilitate the desired scenario application through visual mapping.

The first task in creating this tool, which would separate the elements of the building into four distinct “buckets”, was to assign specific grayscale values to the scenarios. The scale used for this scenario assignment is detailed in Table 9.

Scenario assignment	Black value (%)
SCENARIO A	0-25
SCENARIO B	25-50
SCENARIO C	50-75
SCENARIO D	75-100

Table 9. A scale to determine the grey value assignment to each scenario

The next step was to remap the original elements to fit the image sampler. This was achieved by sorting the input geometries along the X axis, extracting the centroids of all input geometries (after they had been converted to native Breps), and acquiring their XYZ coordinates. These points were then remapped to the native 0 to 1 domain to fit the image sampler (with the rule that the image sampler domain is also set to 0 to 1). Subsequently, all remapped numbers were passed through the image sampler, which assigns a numeric value to these points based on the image input. These remapped points were then sorted according to the scenario thresholds, and the original geometries were dispatched into corresponding data streams.

Regarding the creation of the input image, several approaches were tested. The first approach involved creating a smooth gradient and feeding it into the sampler. The issue encountered with this approach was that the grayscale values falling on the border between scenarios (for example, a value of 0.24 or 0.26) would be assigned unpredictably, depending on the exact pixel location where the remapped element centroid landed.

This prompted a re-evaluation of the grayscale image creation. What was needed for this task was a highly precise, pixelated image - essentially a very low-resolution gradient where the assigned values for each cell are discrete, absolute, and predefined. From this, a pixelated gradient at a resolution of 4x22 pixels (roughly following the structural grid of the building) was tested. This again presented an issue: this image was now too low-resolution to target singular elements for alteration and could work only with the macro-scale cells of existing rooms. Additionally, because the building’s structural grid is irregular, an image resolution this low was deemed unsuitable.

The final solution for the image creation was to evaluate the structural grid and the distances between grid lines (2600 mm and 3200 mm) and to find the greatest common divisor between these numeric values, which is 200. Subsequently, the overall building dimensions were divided by this number, and the final resolution for the grayscale image was determined to be 331x61 pixels. This size allows to have the spatial freedom to target individual elements for alteration while still maintaining a strictly controlled, modular resolution. Practically, the user implements this sampler by underlaying the original floor plan and “painting” these discrete pixels, using an image processing software (for this project Affinity Designer was used) to assign the four scenario values, bridging the gap between the functional diagramming and the algorithmic execution.

While a two-dimensional image may seem like an incompatible input tool for a three-dimensional geometry such as the object building, the decision in favour of using this exact approach has strong justification. First, as described in Section 7, the architectural intention and input always start with the creation of functional diagrams in the plan dimension. Since the workflow aims to

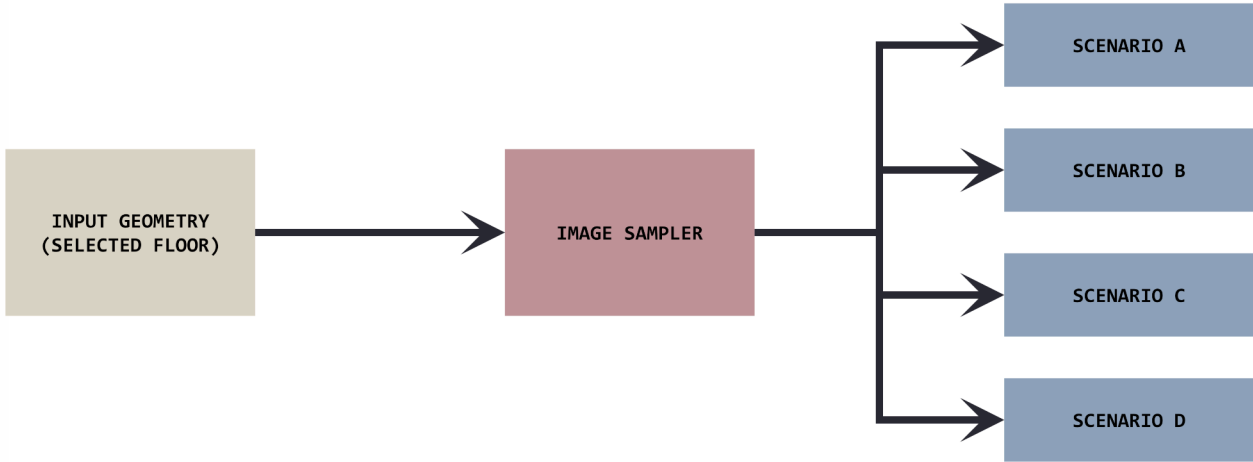


Figure 49. The principle of the image sampler process

reorganise spaces, the approach of always starting with a 2D floor plan and then letting the script handle the three-dimensional generation was logical. Second, the dependency script and mapping operate in whole floor iterations – the floor above will always depend on the floor below Third, the original building consists of five nearly identical floors, therefore, treating them as five identical 2D slices was a decision made early in the project and remained effective throughout the research. Additionally, constraining user input to a two-dimensional format allowed for the simplification of both the creation of the algorithmic script and the user experience. Furthermore, as the tool is intended for the use of architects, reducing the workflow to a two-dimensional image is intuitive, as architects possess the ability to conceptualise in three dimensions nevertheless, and this step allows for the simplification of the workflow without sacrificing the user control. Finally, as this approach proved to work very well from the early stages of the project, the whole system was built around this two-dimensional method, so no further experiments of different approaches were executed.

6.6 THE COMBINED TOOL

Following the development of the modular micro-scripts, the comprehensive generative tool had to be assembled. The scripts were integrated according to the assembly-level scenarios described in Sections 6.3 and 6.4. The geometries sorted through the image sampler were passed into these scenarios, within which the immediate “locked modifications” were executed. The variable geometries were simultaneously passed to the manual decision-making nodes within the script. At the end of a single floor evaluation, the finalised cycle outputs seven distinct data streams: removed blocks, unmodified blocks, removed concrete solids, modified geometry, the original blocks for the modified geometries, the new exoskeleton, and the new endoskeleton (frames).

This is where the dependency script becomes significant. Specific elements from these are extracted and grouped into three foundational containers: [Container 1] unmodified blocks alongside the original blocks for modified geometries; [Container 2] removed blocks from scenario C (interior walls); and [Container 3] removed blocks from scenario D (exterior walls). (The remaining data streams, such as the removed concrete solids and new frames are archived for material quantification and final geometric assembly). Each of these three primary containers is then passed through a separate dependency script. The direct dependencies (iteration 1, structural hierarchy: primary) resting on Container 1 are free to be altered in the next cycle of the next floor. Conversely, the direct dependencies of Container 2 automatically get assigned scenario C in the next cycle (floor). The same happens with Container 3, where all direct dependencies get assigned scenario D in the floor above.

This dependency trigger is the computational backbone of this entire system. The elements dependent on Container 1 are looped back into the primary algorithm, feeding a new image into the sampler for the desired alterations of the next floor. However, the elements dependent on Containers 2 and 3 bypass the image sampler for the next floor, as the scenario of these elements is already predetermined by the scenarios applied to their support elements below.

Similarly, the dependencies on Containers 2 and 3 are calculated for the floor slabs above them. Slabs resting on Container 1 are free to alter, while those resting on Containers 2 and 3 must remain intact to ensure structural continuity over the newly inserted frames. However, following experimental testing of the script (detailed in Section 7), the scenario application on floor slabs was refined, and an independent image sampler was added for this task. The script dictating the floor slab alterations operates parallel to the main script, therefore does not affect the load-bearing walls of the floors above.

The fundamental basis of this system is that the input geometry is one whole floor of the building, and the script then handles the recursive generation of all the floors above it. For the sake of the experiments being comprehensive and clearly showcasing the effect of the dependency calculation, the base input geometry for all experiments is selected as the ground floor, therefore repeating the described process described five times, matching the total number of floors in the object building.

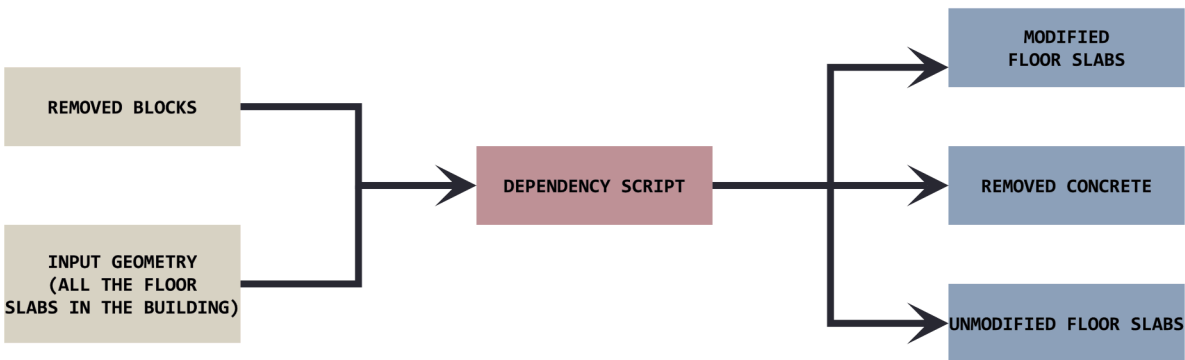


Figure 50. The principle of the recursive dependency script

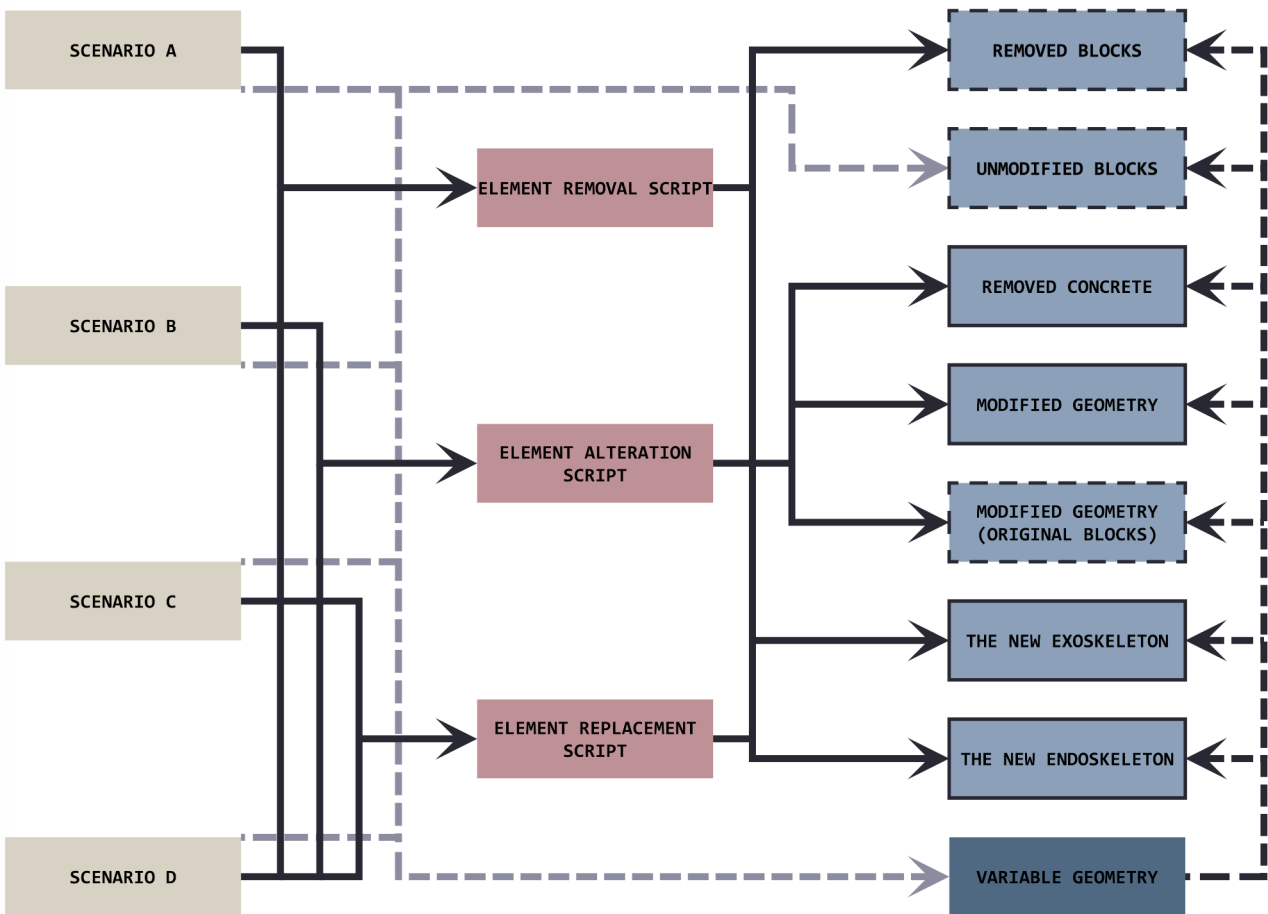


Figure 51. The principle of sorting the geometries into containers

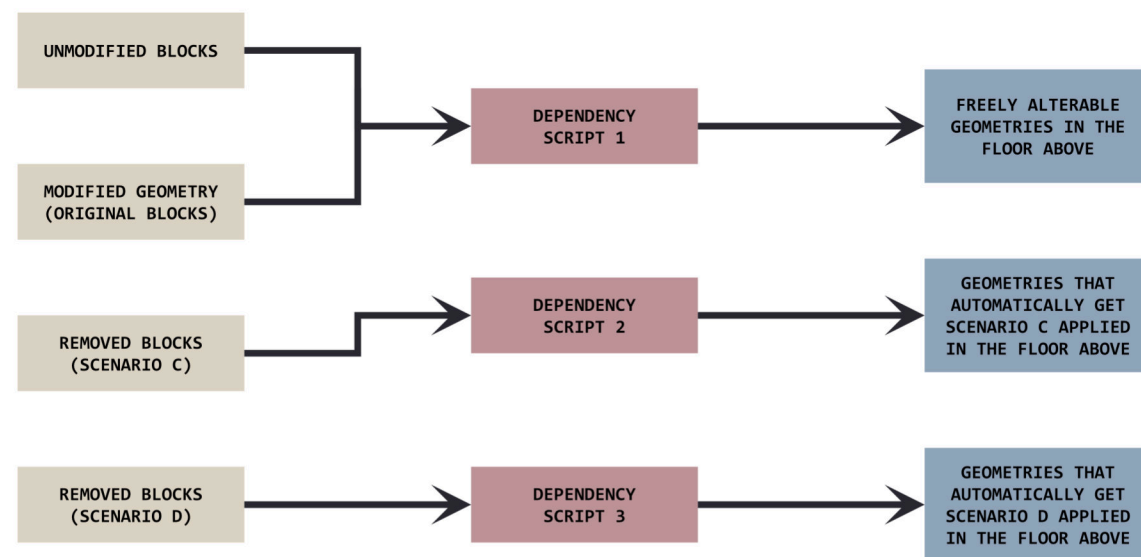


Figure 52. The principle of different dependency script distinctions

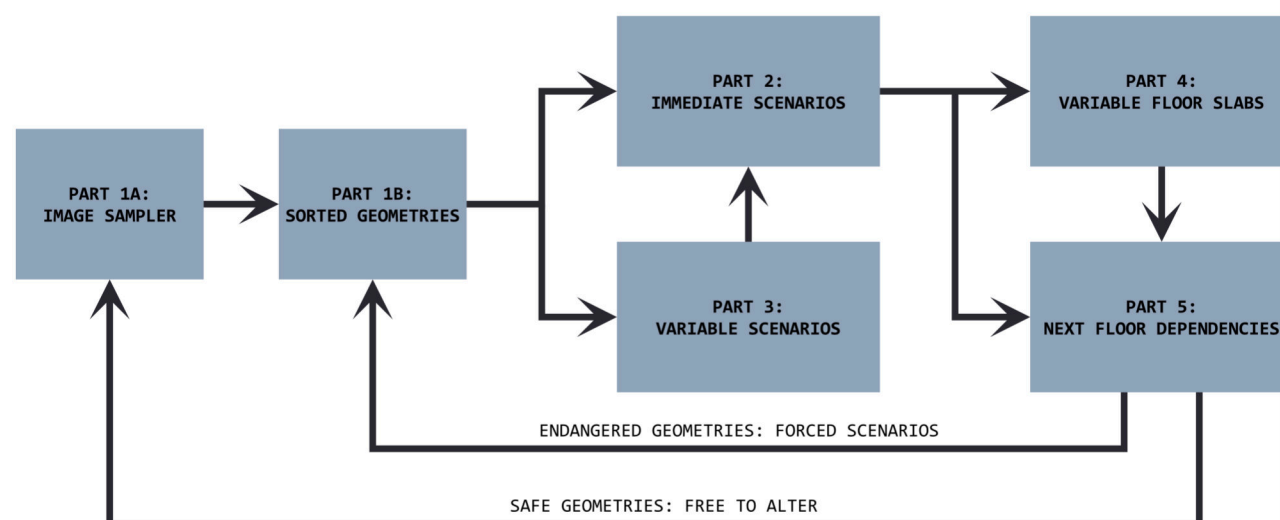


Figure 53. The principle of the script for one whole floor iteration

6.7 USER-GUIDED WORKFLOW AND TOOL INTEGRATION

Evaluating the workflow of the tool through the eyes of a hypothetical architect presents certain challenges, as the system was developed on specific presumptions and logic, therefore, what makes sense to the creator may not immediately make sense to others.

The workflow for an architect using this tool always starts with the original building. Ideally, a digital twin of the building would be created through a combination of the use of precise measurements, 3D scanning or photogrammetry, visual and inventory documentation, as well as original construction drawings. This workflow would ensure that the digital twin precisely reflects the real-world conditions of the building.

The next step is floor plan extraction. This step is essential for the creation of the new functional schemes, new function boundaries and the plumbing nodes that must be preserved. While initially it was hypothesised that the functional injection and zoning diagrams would not depend on the original floor plans, working through the experiments revealed that, as the system aims to reorganise rather than completely rearrange and disrupt the system, it is important to work with the original floor plans. Basing the new functions and layout zoning on the original floor plans also helps to account for the overall dimensions, access to staircases, above-mentioned plumbing nodes etc. Essentially, this is also the step to define which functions are desired to span across more than one floor, to then implement vertical openings in the floor slabs.

The subsequent step is the translation of the functional diagrams to the grayscale images for the image sampler. While it may seem logical to just sketch the grayscale images from the start, it is essential to separate these two steps. As found out in the experiments, the functional injection diagrams help to identify the boundaries of new spaces and identify what types of spaces are needed within each, marking this as a “building scale” step; the grayscale image actually works more in the “element scale”. In the creation of the grayscale image, the architect looks at the functional diagrams alongside the original floor plan, and identifies what is required for

each element scenario to be. This is the moment where user experience becomes essential, because, while in the first experiments the scenarios were assigned to whole spaces, later iterations demonstrated that assigning them to individual elements offers way more control over the process (see Section 7).

One possible approach to this translation process is the workflow proven to be most effective: first, the boundaries of all the new spaces are identified from the functional diagrams and assigned scenario A (white). Then, the outer boundaries (the exterior walls) within these functions are analysed based on assumed need (e.g., which ones need full height glazing, which can remain as they are, which require full removal of the wall and the expansion towards the exoskeleton) and the pixels are assigned accordingly. Finally, the partitions and walls within the new functions are assigned scenarios, again based on assumed need (for example, if a fully open-layout is needed for the function, scenario C – dark grey – is assigned). Additionally, for the functions that have been defined to be two or more stories, an additional grayscale image is created, choosing where to remove or alter the floor slabs.

To reflect on “how would it feel for an architect to use this tool” - it does take some getting used to. It may not be immediately intuitive, and it is definitely not random. The output highly relies on the combination of user input, dependency mapping and the forced scenario application that results from it. Again, it is a bit difficult to reflect on this objectively from the creator’s standpoint, since the exact trigger points for the dependency script and forced scenarios are already known. Therefore, the best way to get acquainted with the tool is through experiments and tests. However, the best workflow identified for the translation of the functional diagrams to the grayscale image is underlaying the original floor plan, and, while looking at the functional diagrams, “painting the pixels” and deciding what should happen to each element.

7. ITERATIVE TESTING AND SCENARIO EVALUATION

This section evaluates the application of the algorithmic tool through a series of computational experiments. These tests were conducted to evaluate system's performance, discover its flaws, and improve the flow of the user-guided process to enable optimal results. This chapter concludes with one final architectural output as proof that the system works and to serve as an example of the infinite structural and spatial alteration possibilities unlocked by the created framework.

7.1 EXPERIMENT 1

Experiment 1 was conducted to test the baseline applicability of the tool. The ground floor of the building was used as the input geometry for the start of the algorithm. As the grayscale image sampler input, a generic, pixelated gradient was created (with a resolution of 22 x 4 pixels, see Section 6.5). The image is hardly definable as a gradient, as it was purposely created to showcase the four distinct scenario colours and to restrict “on the border” values that could be falsely interpreted by the image sampler. This process was conducted in Affinity Designer software, using a pixel brush tool to ensure sharp boundaries between the assigned scenarios. This experiment, however, was conducted without an architectural or spatial intent; i.e. the image for the sampler was created without direct connection to the existing floor plan. Instead, the experiment was executed as an initial test to verify if the tool works as intended.

The pixelated gradient image was then used to assign the scenarios to the input ground floor geometries (see Figure 57, Step 1). The script was then run to visualise the immediate scenario applications, as well as the elements were assigned “variable” scenarios (Figure 57, Step 2) (see Section 6.4). The scenarios for the variable elements were then chosen using the decision-making inputs of the script and applied to the elements (Figure 58, Step 3). As a final step for a whole floor cycle, the scenarios for the floor slabs above the floor were assigned (Figure 58, Step 4). At this stage of the script, the floor slab scenario determination was as follows: all floor slab panels resting atop removed/replaced walls were assigned as “unmodified”, while the ones resting atop unmodified and modified (partial removal) walls were assigned as “modifiable”. For the latter, the decision could be made within the script whether to perform a partial removal or leave them unmodified. This approach was reevaluated in later experiments due to offering little to no control over the new connections in the vertical dimension. Finally, the removed concrete and whole elements were accounted for (Figure 58, Step 5). This marks the completion of one cycle.

To start the next cycle, the dependency script had to be deployed. The removed (replaced) elements from cycle 1 get passed on to two distinct dependency scripts and the

direct dependencies on these elements get automatically assigned respective scenarios C and D in cycle 2 (Figures 58 and 59, Steps 6 and 7). All other elements (unmodified and modified) from cycle 1 get passed through another separate dependency script to determine the “free to alter” elements for cycle 2. Finally, a new image sampler is applied to the next floor above (2nd floor). A reversed gradient was chosen for this step to showcase the dependency trigger. Then, steps 1-5 are repeated and an output is generated. Output 9 (Figure 59, Step 9) clearly shows the forced scenario application.

This experiment was concluded here, as it helped to identify functional issues and there was no need to finalise the whole building design given that the inputs were created with no architectural intention.

Note: the removed elements in Figures 57, 58, and 59 do not show the removed WCx blocks due to their structural insignificance, low reuse potential, and the complications of integrating these elements within the script.



Figure 54. The image applied to the image sampler on the original input geometry

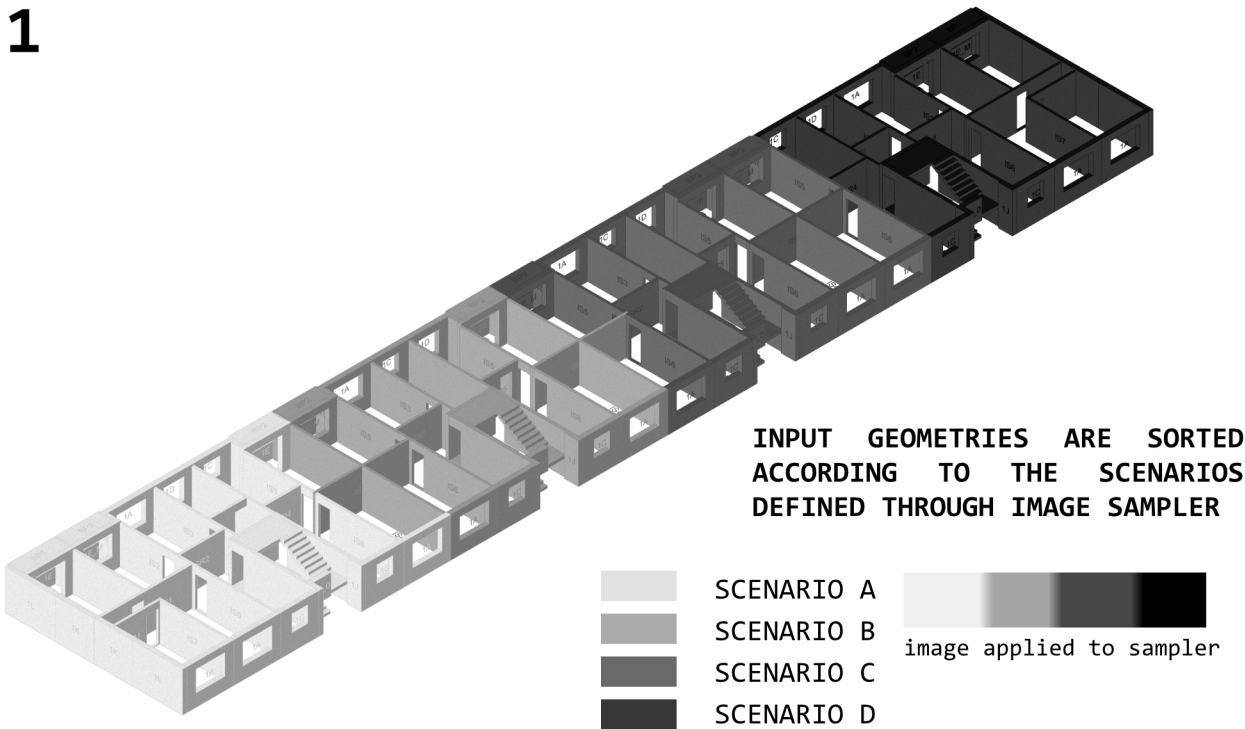


Figure 55. The image applied to the image sampler on the 2nd floor



Figure 56. The adjusted image after the dependency script is triggered and forced scenarios are applied

1



2

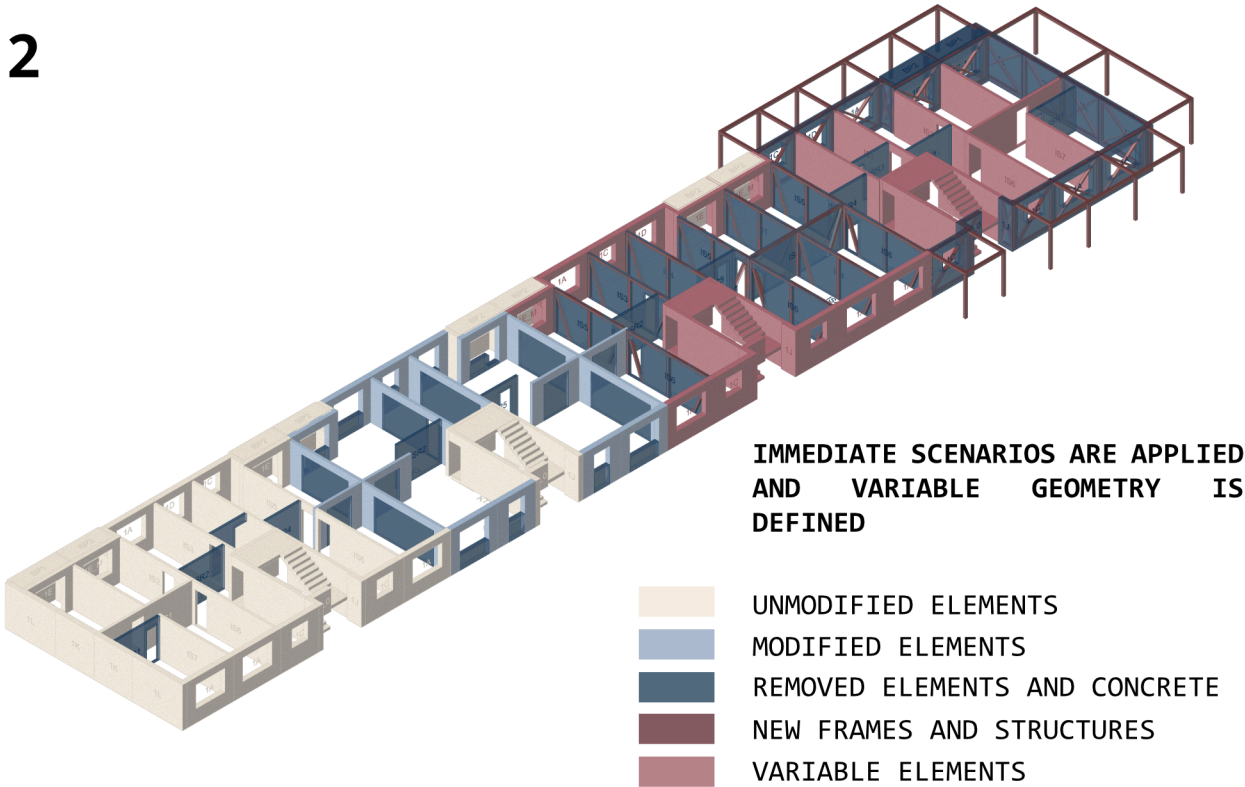
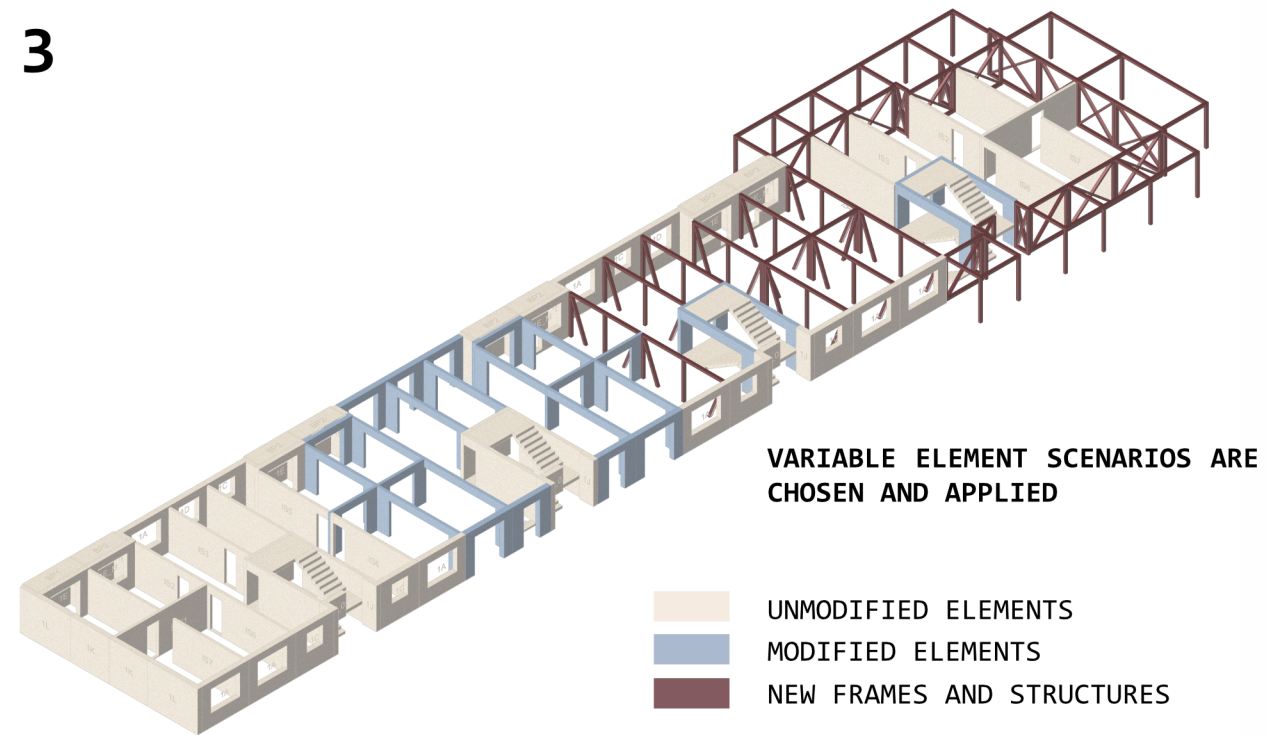
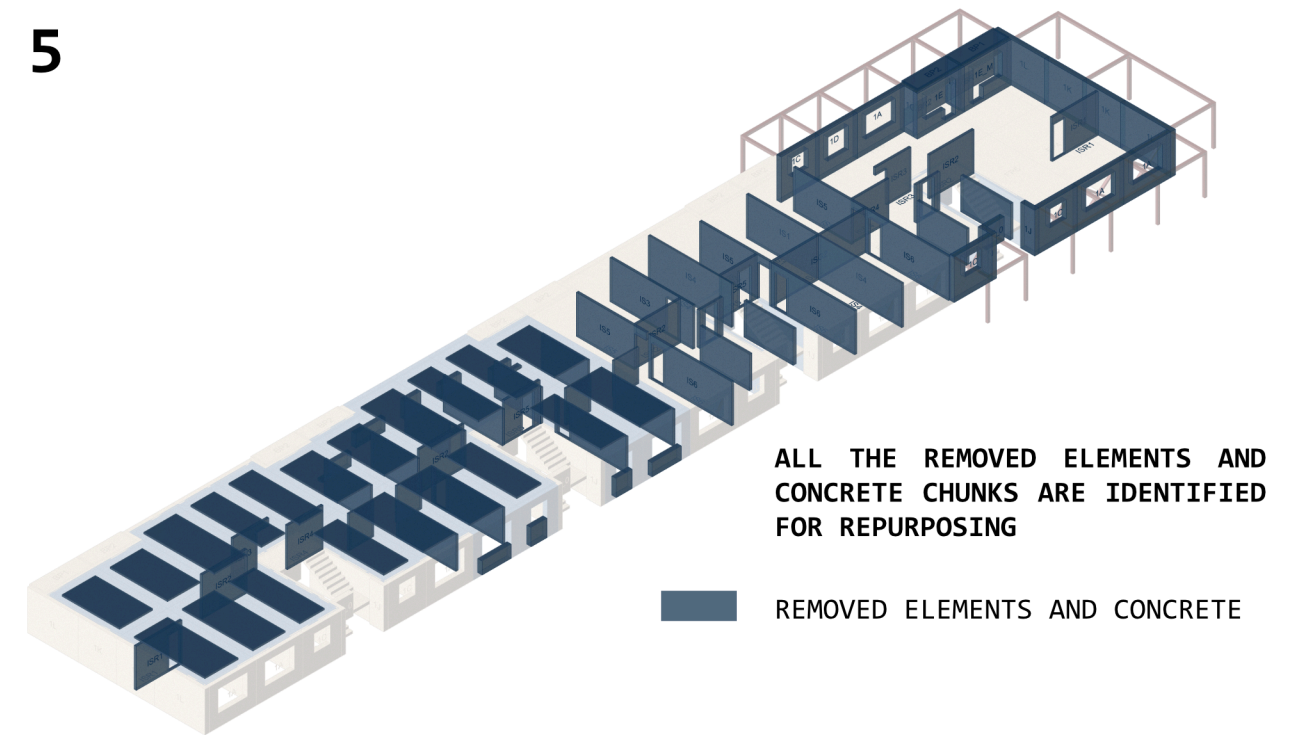


Figure 57. Experiment 1, part 1

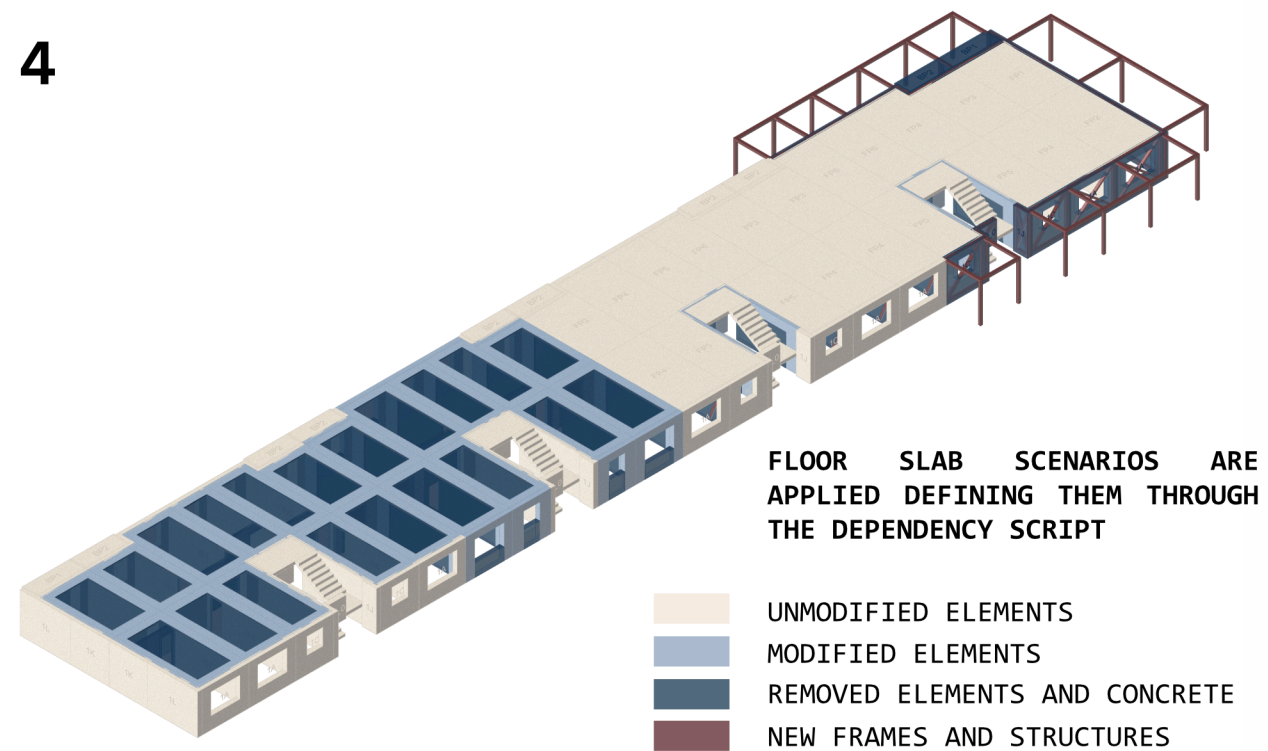
3



5



4



6

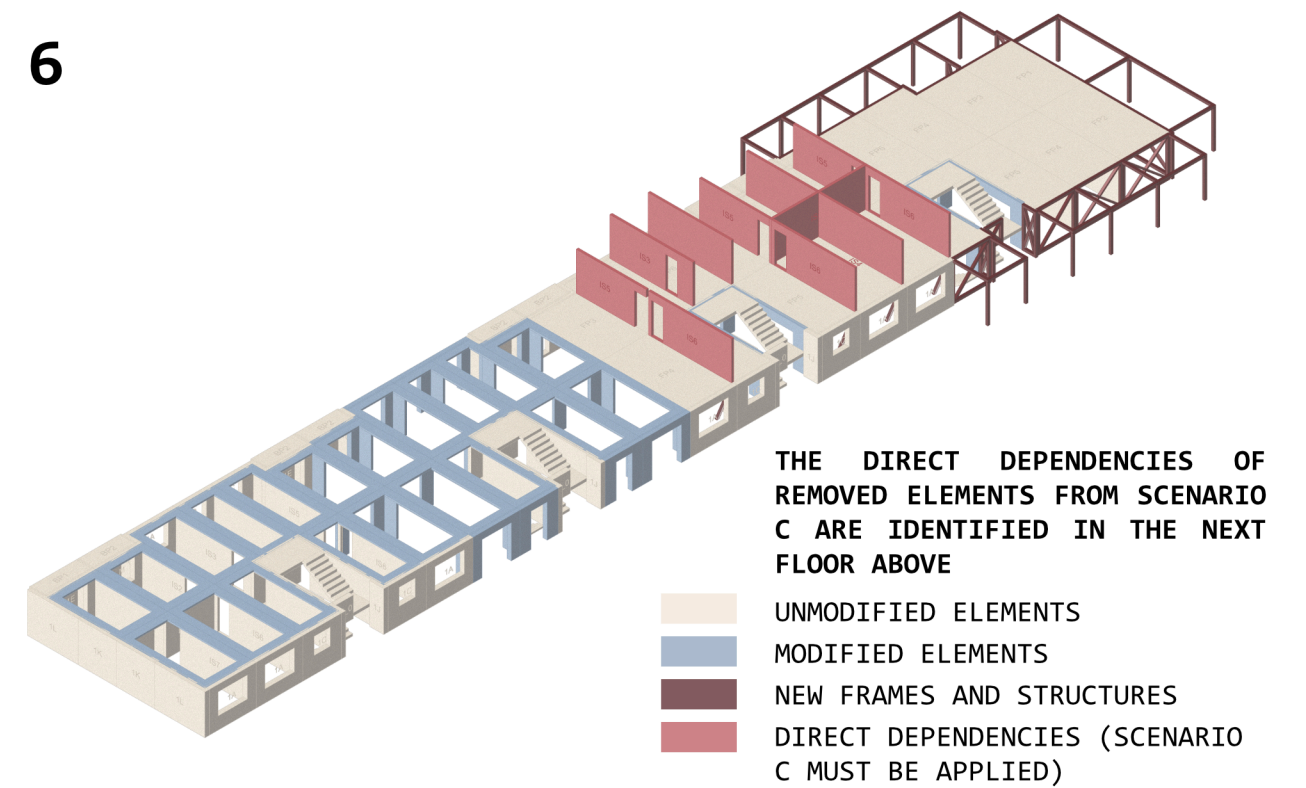
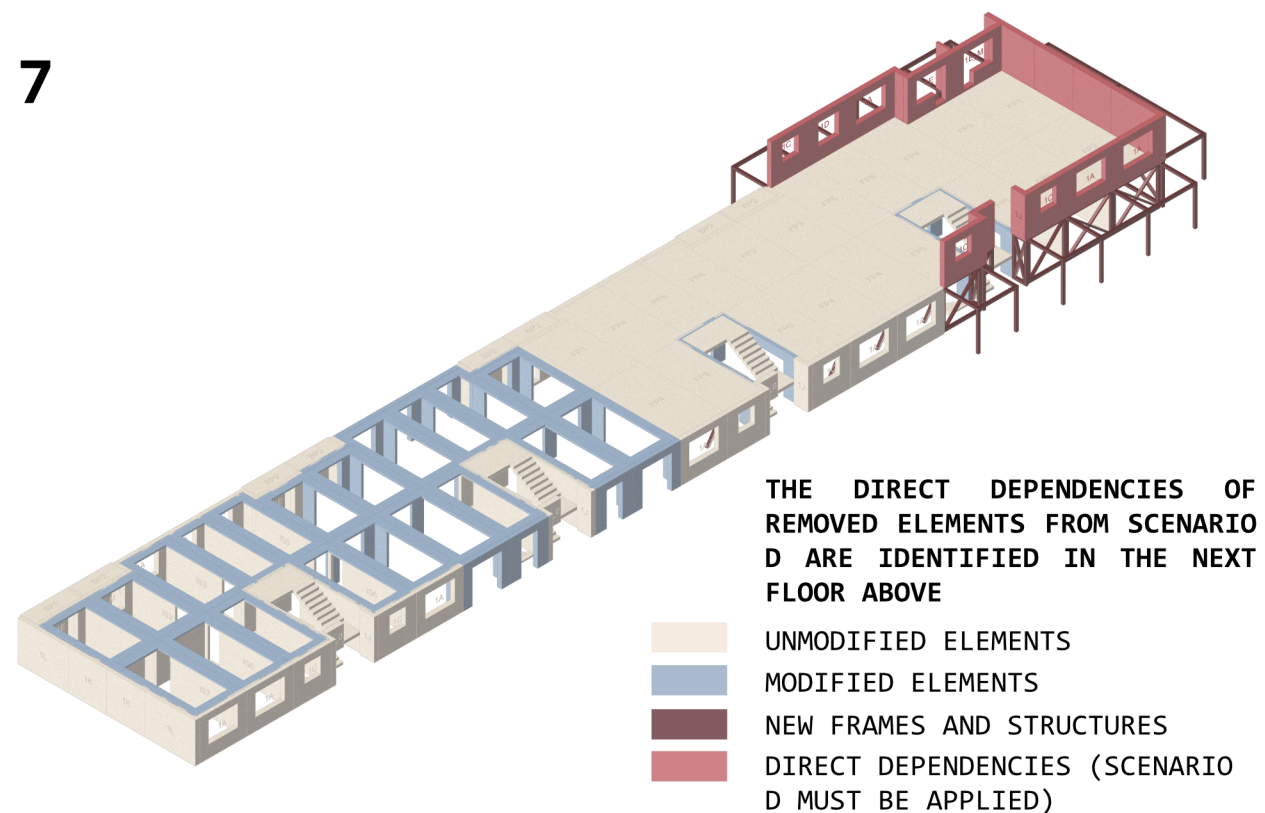
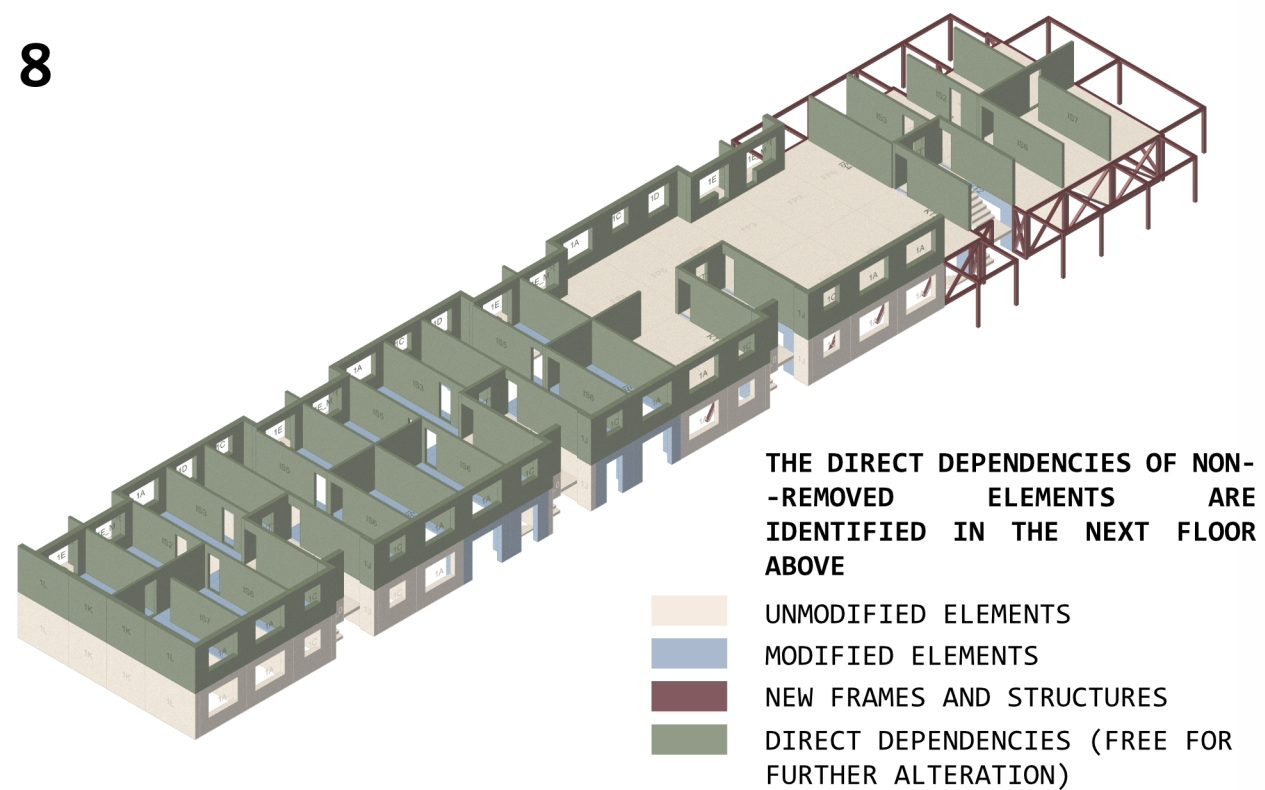


Figure 58. Experiment 1, part 2

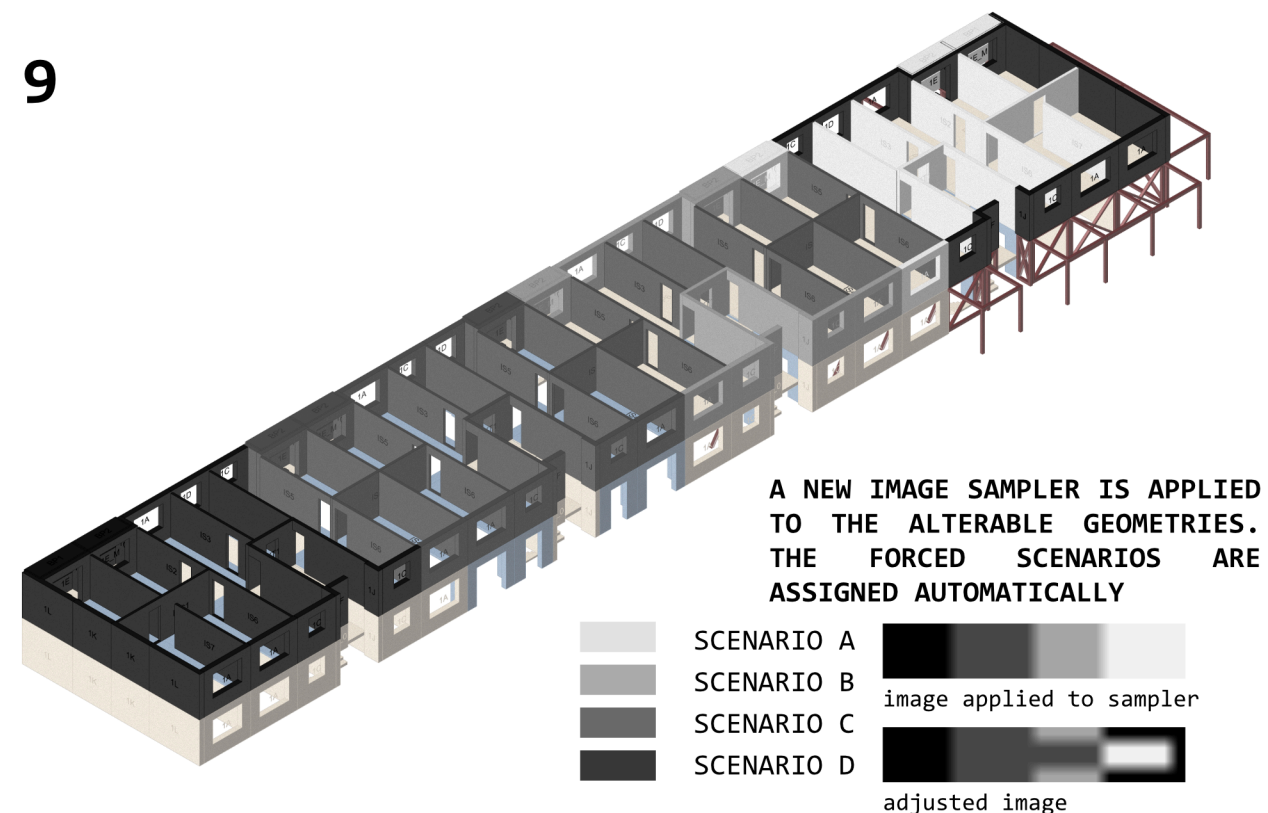
7



8



9



10

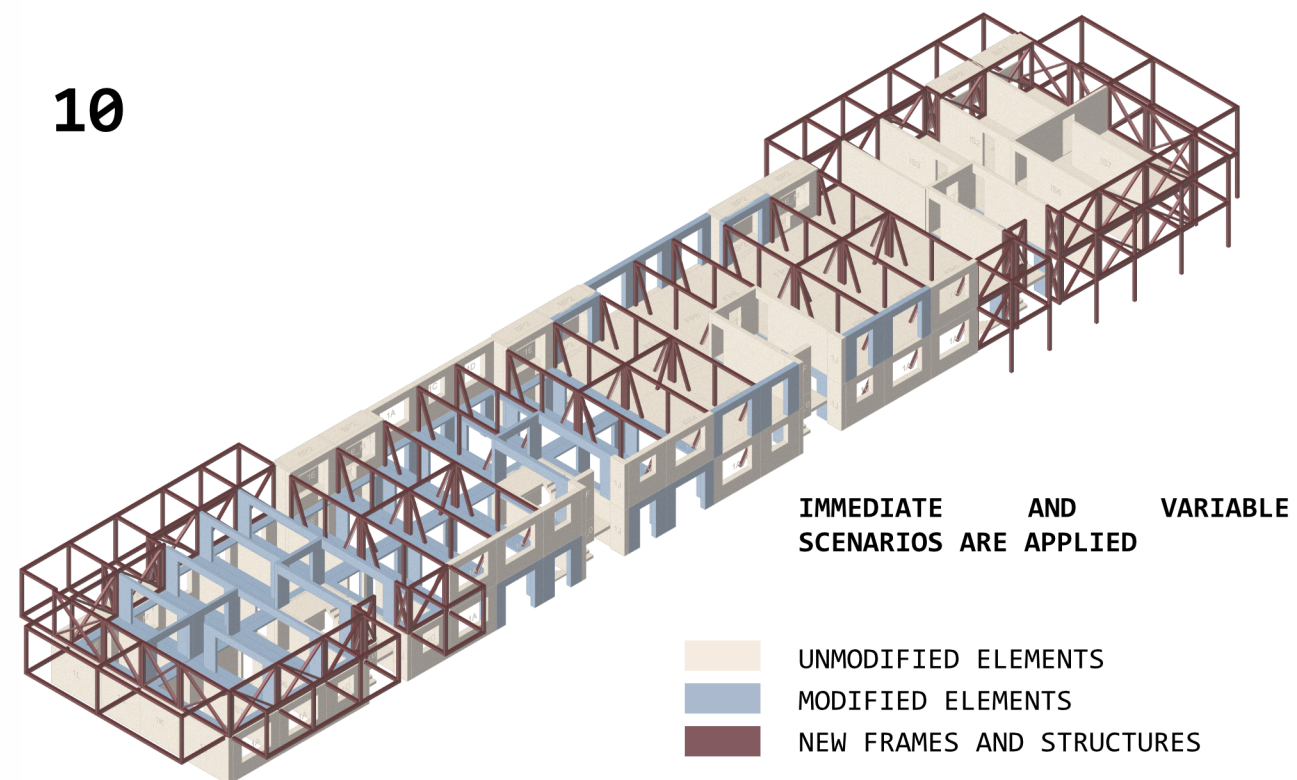


Figure 59. Experiment 1, part 3

7.2 EXPERIMENT 2

Experiment 2 was again conducted to test the tool's functionality. In this iteration, the generation of the building across its entire height was tested. The focus in this experiment was on showcasing the working dependency application, where only a small section of each floor was initially targeted to have Scenario D applied, but the adjusted output forced half of the building into this scenario. This test was also conducted without a specific architectural intention.

The iterations through the floors are not showcased in Figures 61 and 62, because the focus of this experiment was the application of Scenario D, where the main subject is the removal and replacement of exterior wall panels. As can be seen in Figure 62, Step 2, the exoskeleton generation script successfully detects where it should be applied; however, the process does not calculate downward loads. Because the generative script runs from the bottom floor upwards and does not loop back to previous steps, it does not detect that if the exoskeleton is generated on, for example, the 3rd floor, it should also be generated on the ground floor and 2nd floor. However, this flaw

was already accounted for, and a decision was made not to fix it within the script, but rather leave it as a post-algorithmic step for the user to detect and alter. Alternatively, an option to utilise this script to consider elements from Scenario D as completely removed was considered within this experiment (Figure 62, Step 3), but was discarded from further development as it does not comply with the created system and the desired architectural outputs.



Figure 60. Original applied gradients next to gradients adjusted by forced scenario application

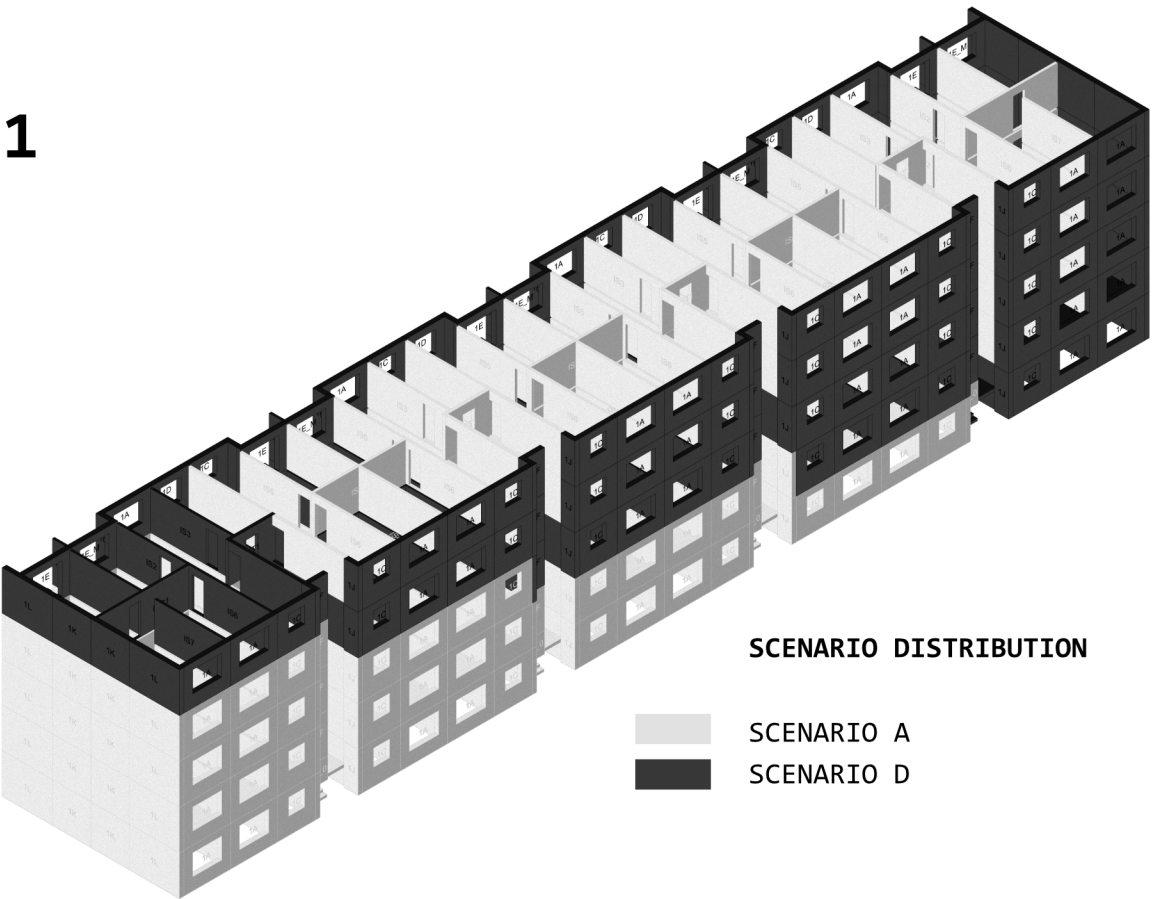
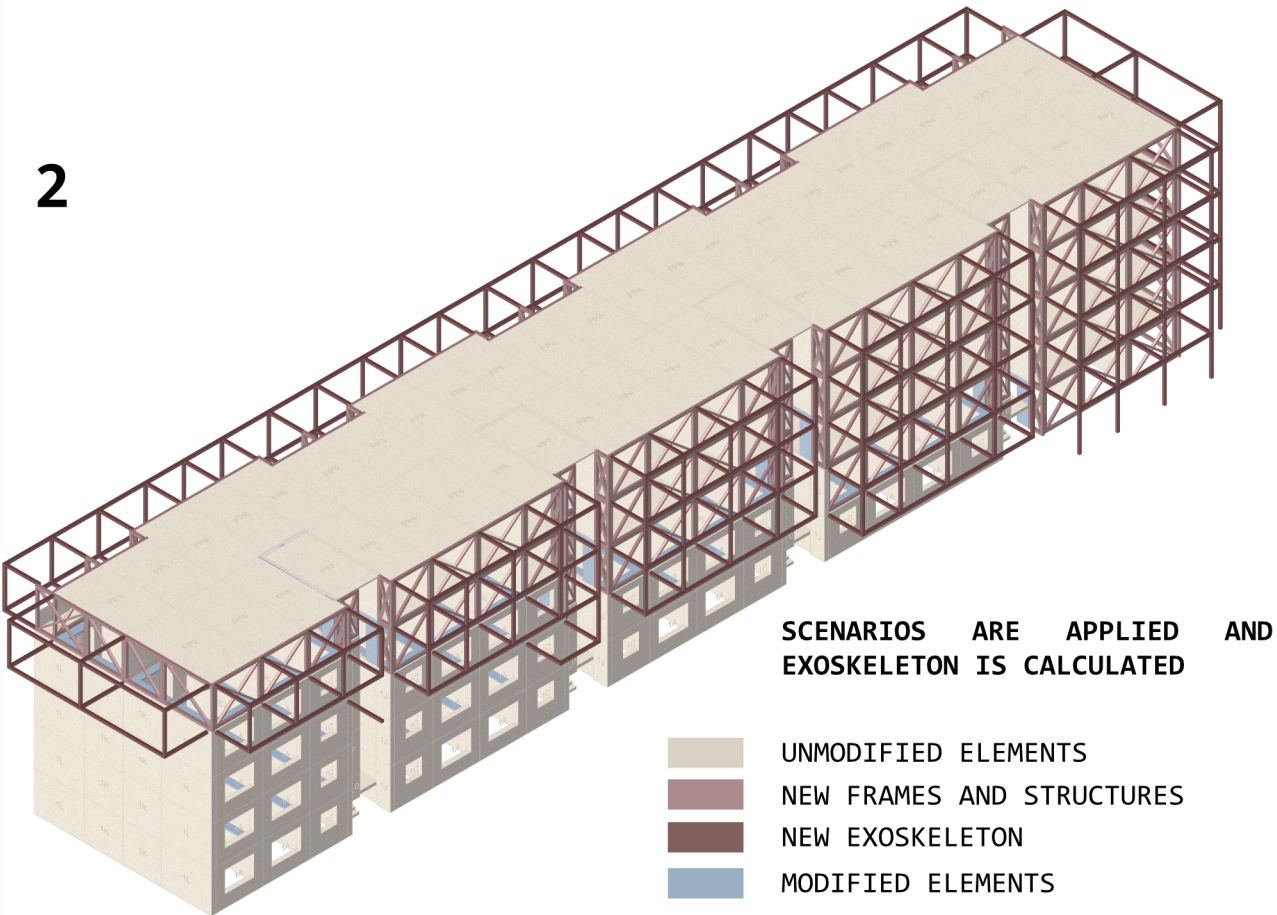


Figure 61. Experiment 2, part 1: full height scenario application testing

2



3

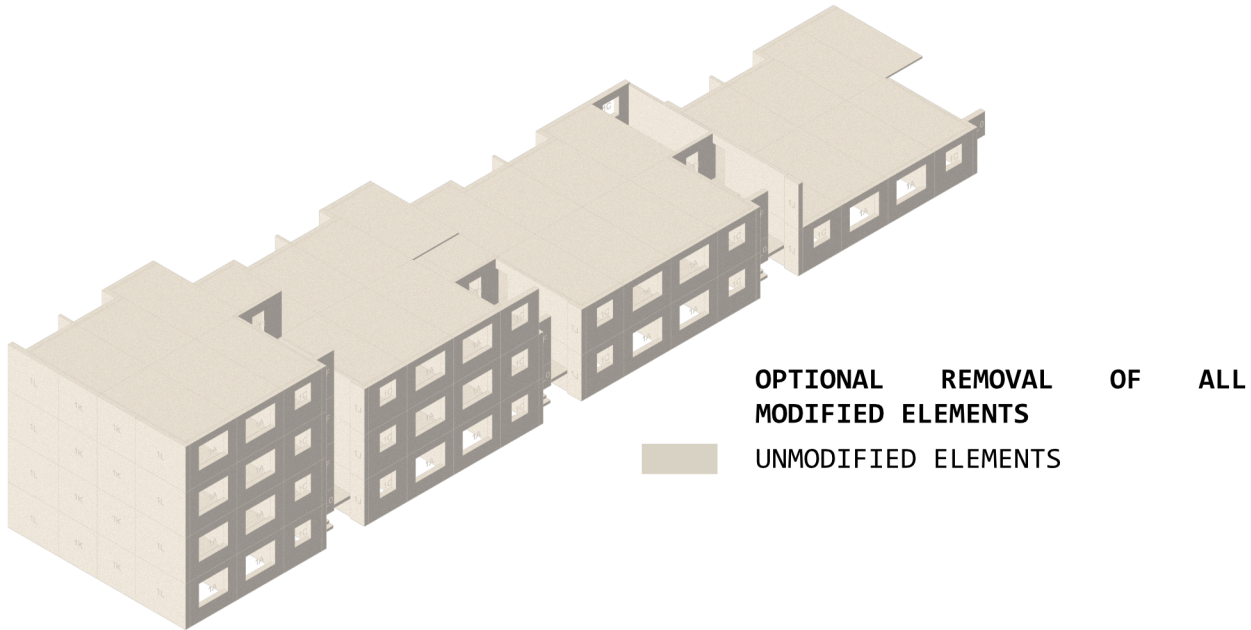


Figure 62. Experiment 2, part 2: exoskeleton calculation and optional geometry removal

7.3 EVALUATING THE SPATIAL CONSEQUENCES OF EACH SCENARIO

Since the first two experiments evaluating the tool were successful, it was necessary to determine each scenario's programmatic suitability and spatial consequence before running subsequent tests with more defined architectural intention and functional injection. The scenarios were already established in

Section 6, but their spatial consequences had yet to be evaluated. This was achieved by analysing the scenarios in the floor plan dimension. These spatial evaluations are represented in Figures 63 to 66. As evaluated in the diagrams, the spatial outcomes scale directly with the level of intervention - ranging from minor layout alterations within the existing apartments in Scenario A to complete removal of exterior envelope for an outwards floor plan extension in Scenario D.

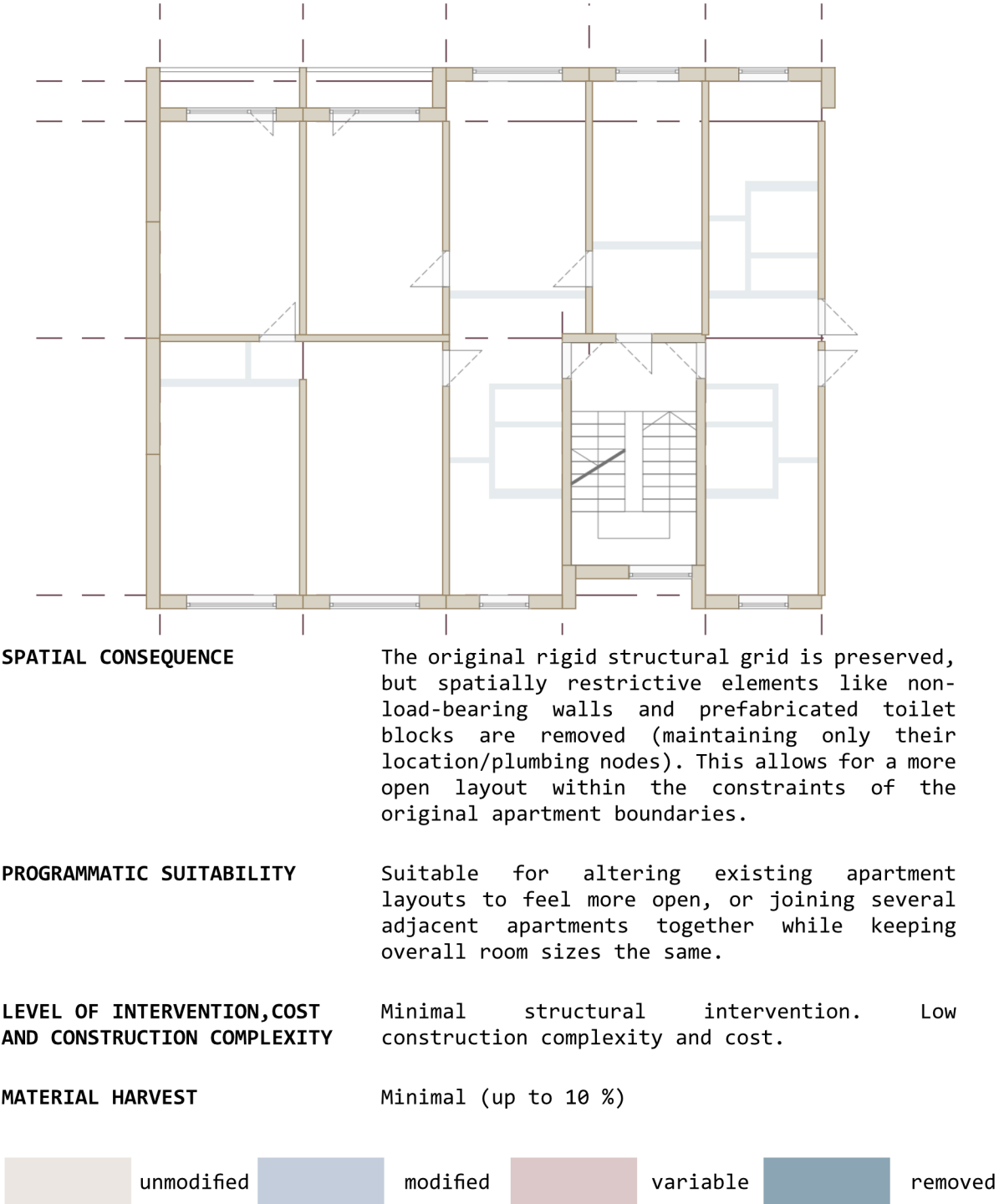
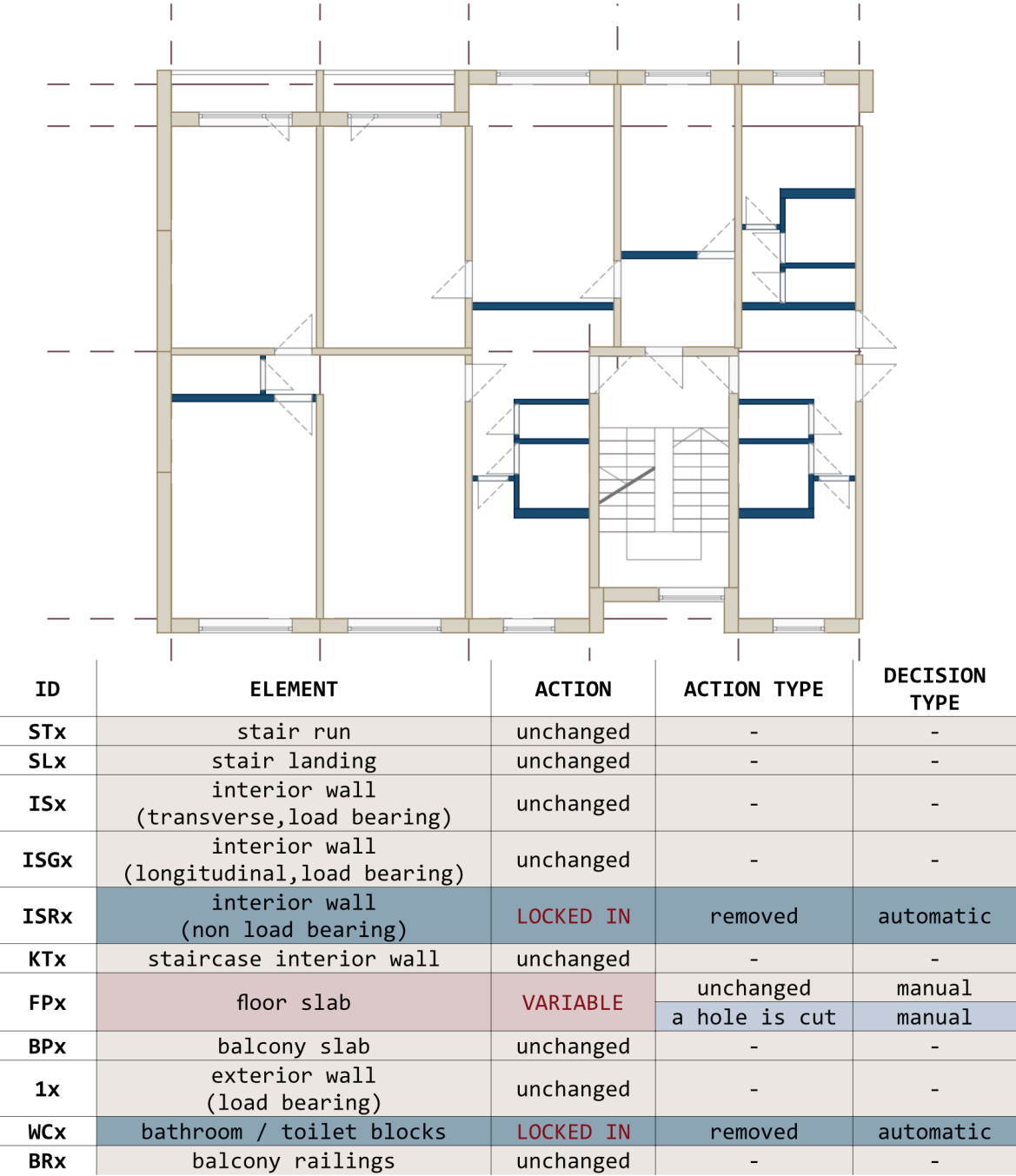
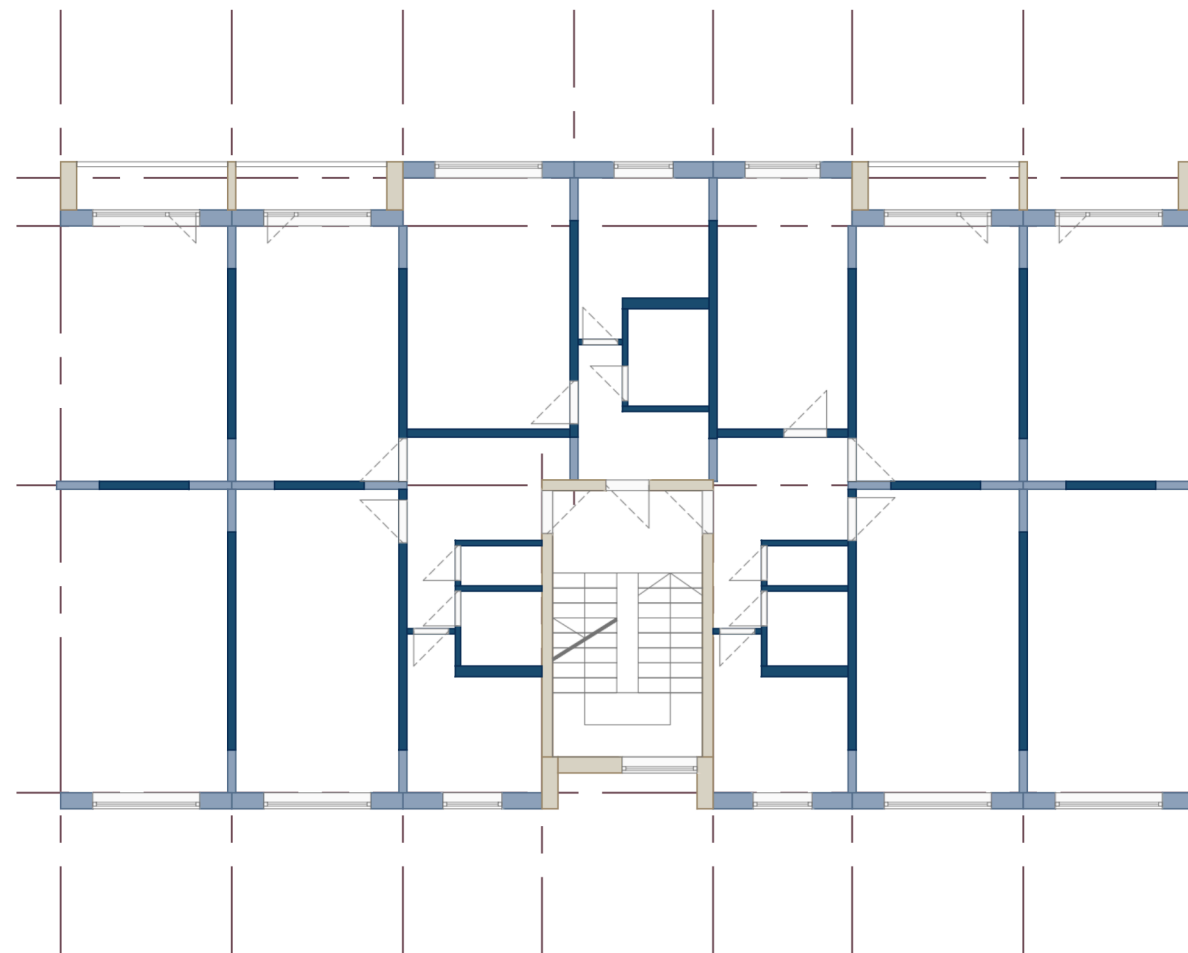
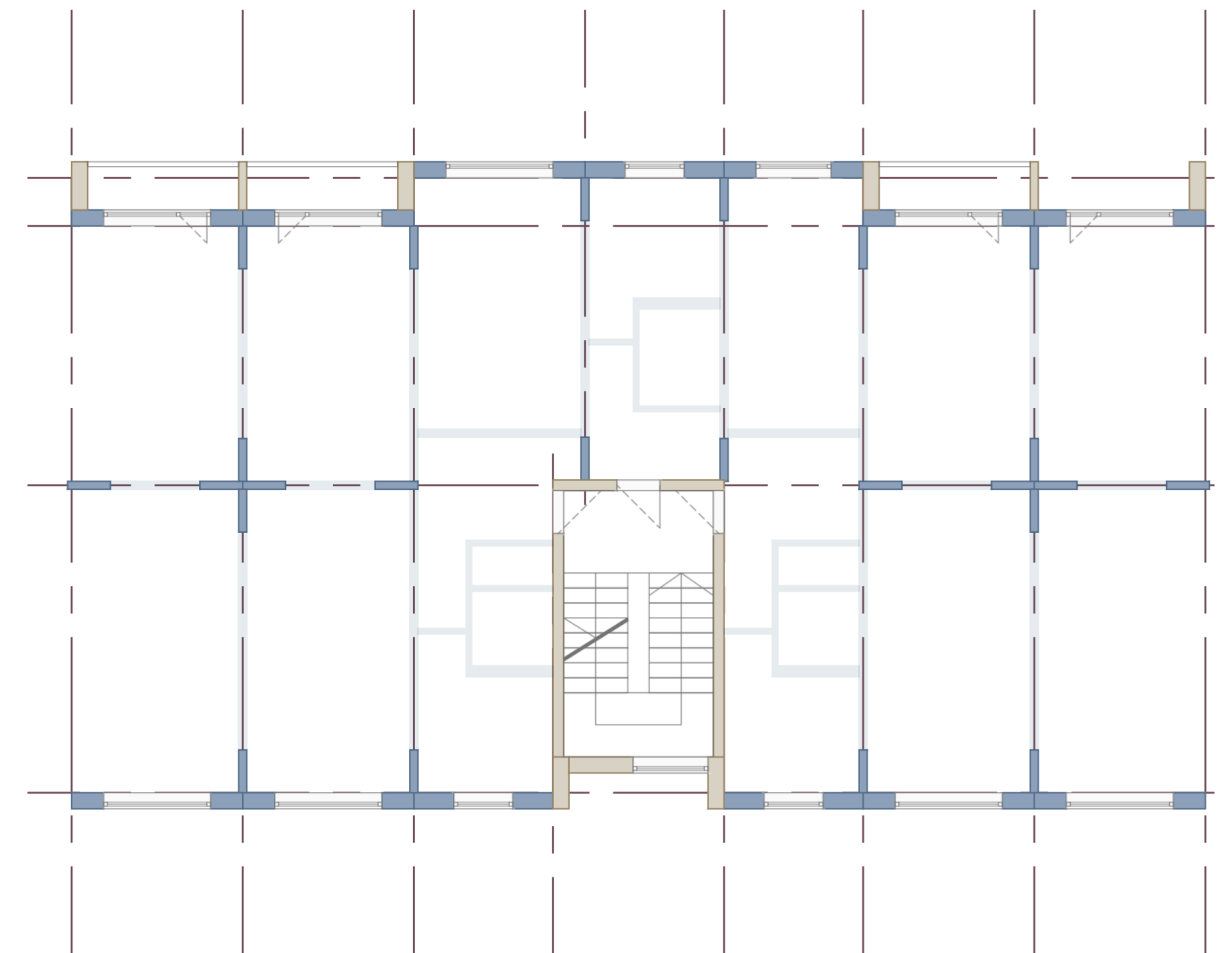


Figure 63. Scenario A spatial and programmatic evaluation



ID	ELEMENT	ACTION	ACTION TYPE	DECISION TYPE
STx	stair run	unchanged	-	-
SLx	stair landing	unchanged	-	-
ISx	interior wall (transverse,load bearing)	LOCKED IN	a hole is cut	automatic
ISGx	interior wall (longitudinal,load bearing)	LOCKED IN	a hole is cut	automatic
ISRx	interior wall (non load bearing)	LOCKED IN	removed	automatic
KTx	staircase interior wall	unchanged	-	-
FPx	floor slab	VARIABLE	unchanged	manual
BPx	balcony slab		a hole is cut	manual
1x	exterior wall (load bearing)	LOCKED IN	existing opening is expanded	automatic
WCx	bathroom / toilet blocks	LOCKED IN	removed	automatic
BRx	balcony railings	unchanged	-	-



SPATIAL CONSEQUENCE

The primary grid and spatial boundaries are preserved to some extent, but partial wall removals open up possibilities to combine rooms. Additionally, existing openings in the external walls are expanded downwards, introducing more daylight and establishing new outdoor connections.

PROGRAMMATIC SUITABILITY

Suitable for creating large, open-layout apartments or introducing new public spaces where room sizes need to be varied.

LEVEL OF INTERVENTION,COST AND CONSTRUCTION COMPLEXITY

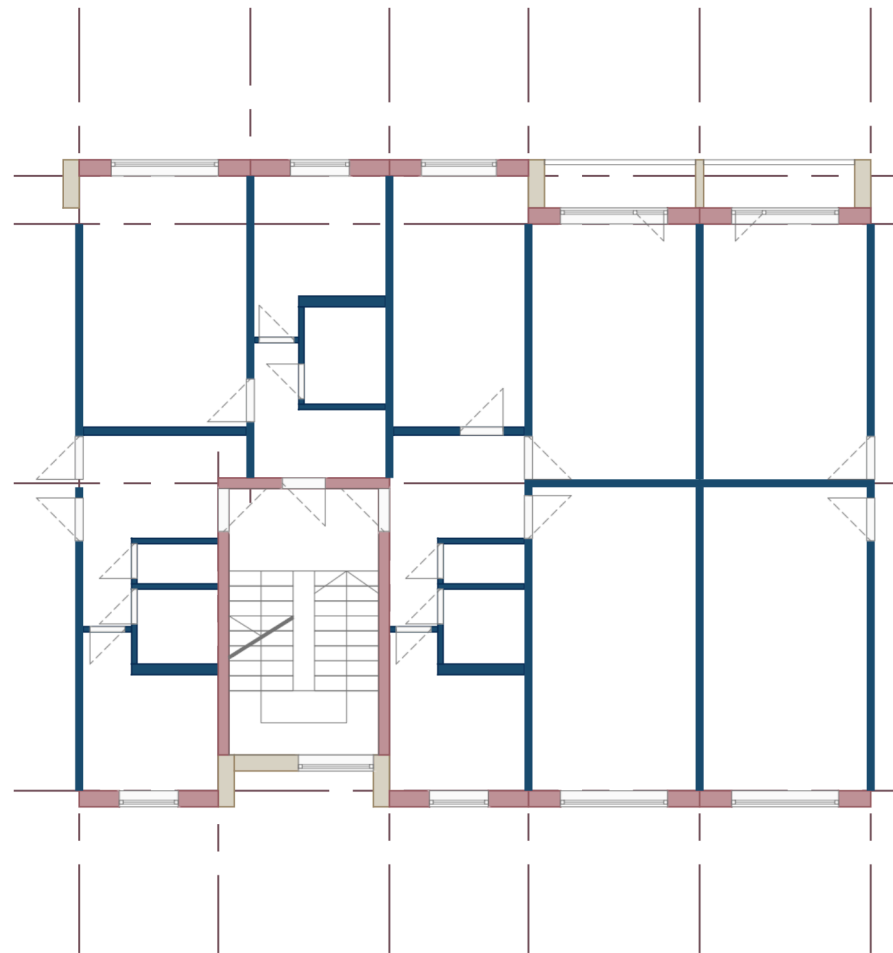
Moderate structural intervention. Medium complexity and cost, requiring precise concrete cutting and localised frame reinforcements.

MATERIAL HARVEST

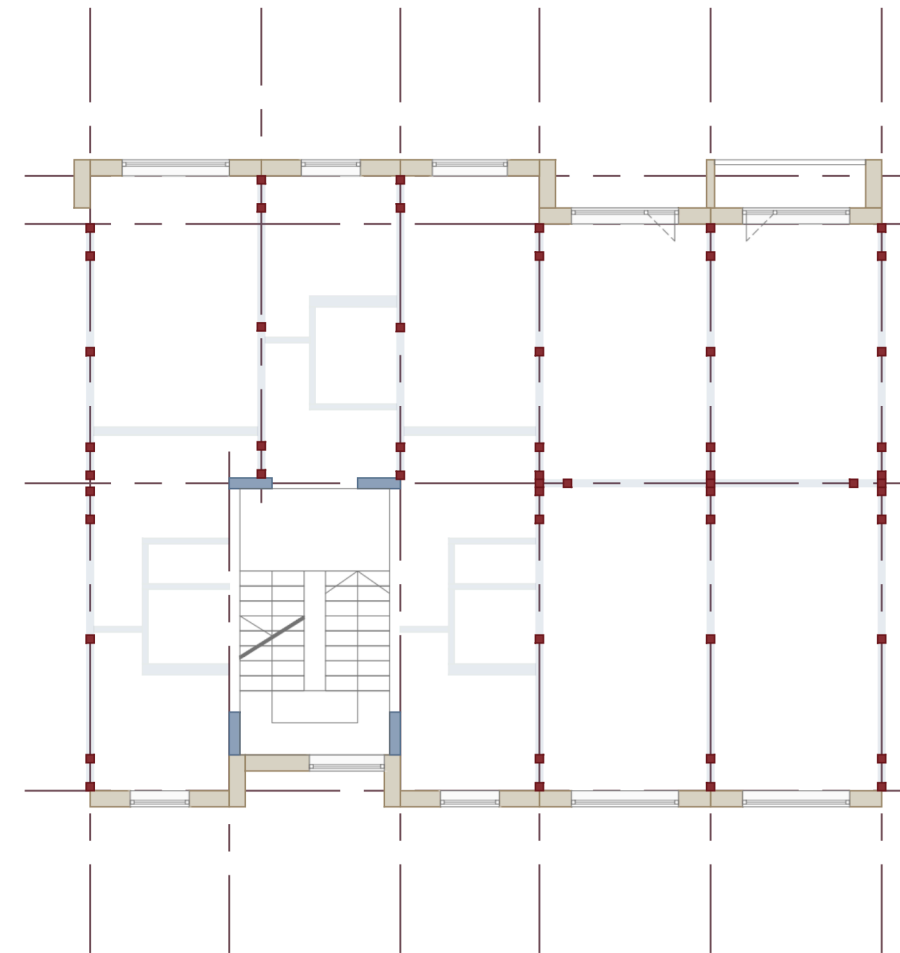
Moderate (up to 30 %)



Figure 64. Scenario B spatial and programmatic evaluation



ID	ELEMENT	ACTION	ACTION TYPE	DECISION TYPE
STx	stair run	unchanged	-	-
SLx	stair landing	uncganged	-	-
ISx	interior wall (transverse,load bearing)	LOCKED IN	a hole is cut	automatic
ISGx	interior wall (longitudinal,load bearing)	LOCKED IN	a hole is cut	automatic
ISRx	interior wall (non load bearing)	LOCKED IN	removed	automatic
KTx	staircase interior wall	VARIABLE	unchanged	manual
			a hole is cut	manual
			replaced by frame	manual
FPx	floor slab	VARIABLE	unchanged	manual
			removed	manual
BPx	balcony slab	unchanged	-	-
1x	exterior wall (load bearing)	VARIABLE	unchanged	manual
			existing opening is expanded	manual
WCx	bathroom / toilet blocks	LOCKED IN	removed	automatic
BRx	balcony railings	unchanged	-	-



SPATIAL CONSEQUENCE

The structural grid is maintained, but solid concrete walls are entirely replaced by lightweight spatial frames. The physical barriers between rooms are dissolved, creating possibilities for larger spaces,

PROGRAMMATIC SUITABILITY

Best utilised for introducing public functions or creating spacious, unpartitioned open-layout apartments.

LEVEL OF INTERVENTION,COST AND CONSTRUCTION COMPLEXITY

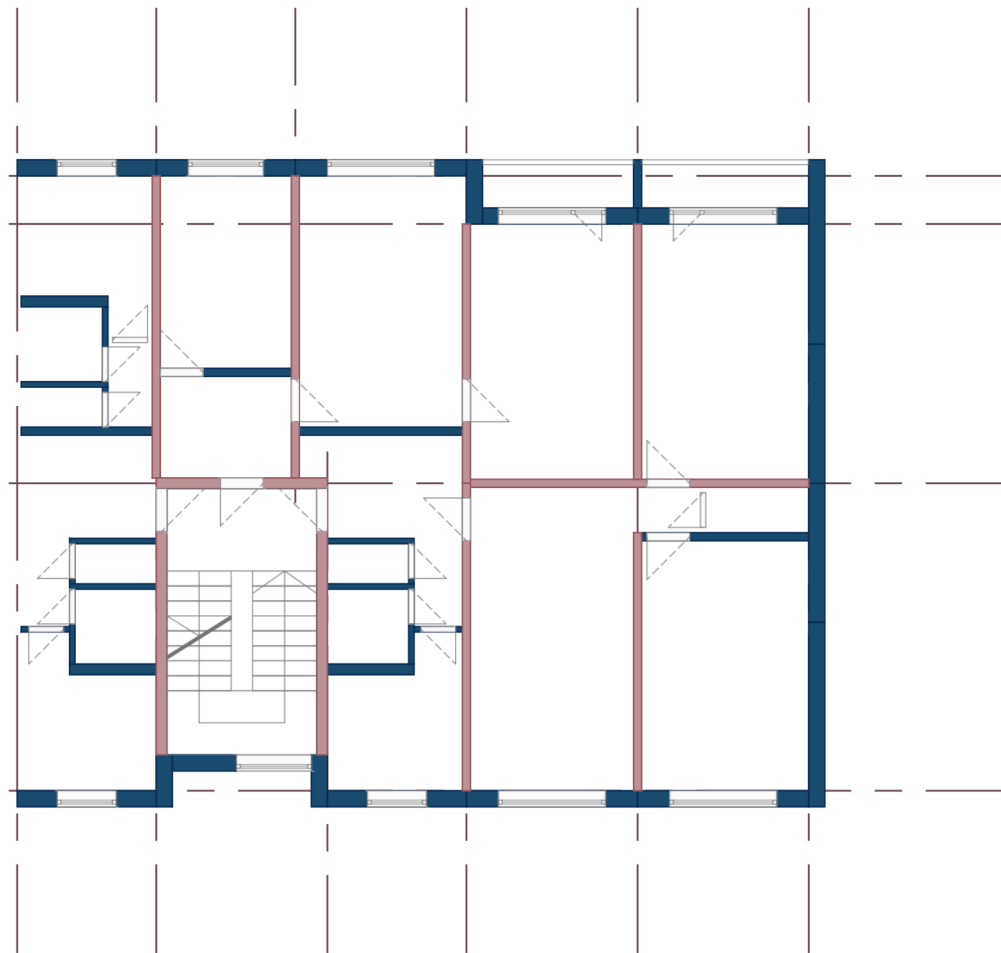
High structural intervention. Significant complexity and cost due to extensive steel frame insertions and heavy load transfers.

MATERIAL HARVEST

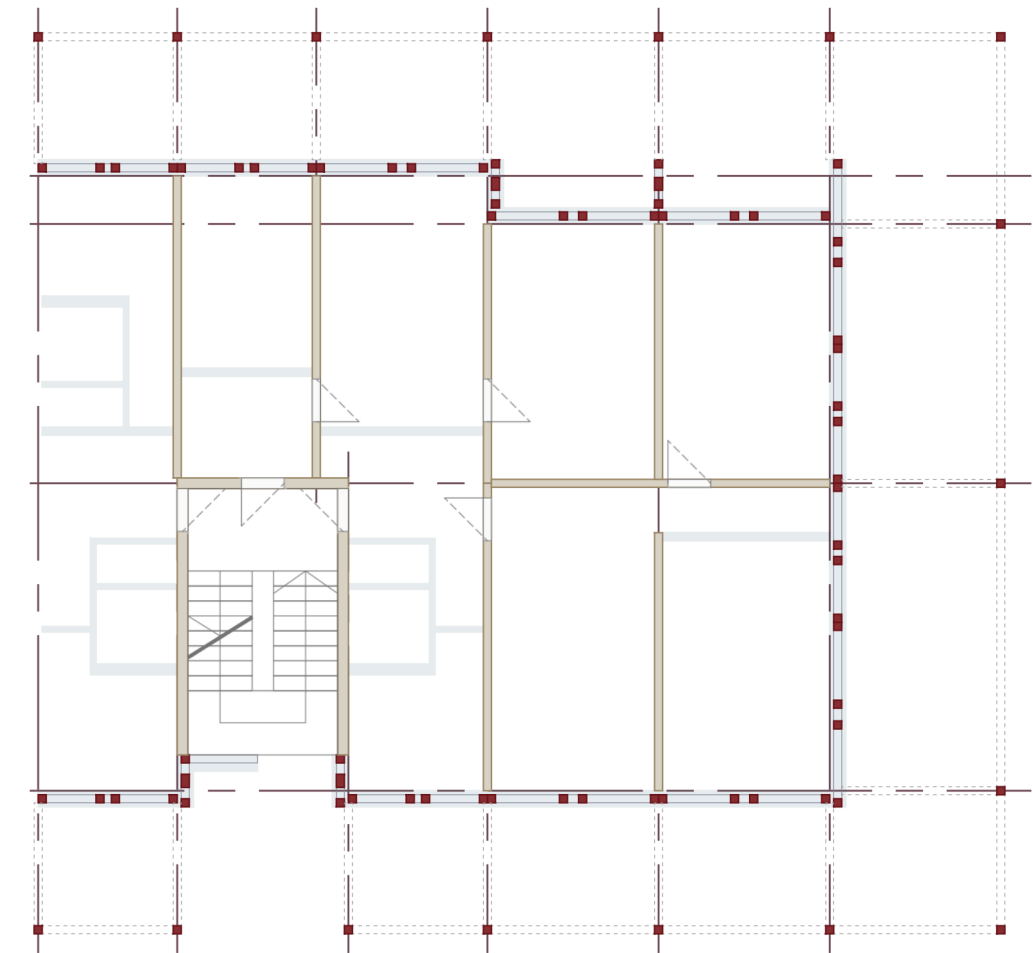
High (up to 50 %)



Figure 65. Scenario C spatial and programmatic evaluation



ID	ELEMENT	ACTION	ACTION TYPE	DECISION TYPE
STx	stair run	unchanged	-	-
SLx	stair landing	unchanged	-	-
ISx	interior wall (transverse,load bearing)	VARIABLE	unchanged	manual
			a hole is cut	manual
			replaced by frame	manual
ISGx	interior wall (longitudinal,load bearing)	VARIABLE	unchanged	manual
			a hole is cut	manual
			replaced by frame	manual
ISRx	interior wall (non load bearing)	LOCKED IN	removed	automatic
KTx	staircase interior wall	VARIABLE	unchanged	manual
			a hole is cut	manual
			replaced by frame	manual
FPx	floor slab	VARIABLE	unchanged	manual
			removed	manual
BPx	balcony slab	LOCKED IN	removed	automatic
1x	exterior wall (load bearing)	LOCKED IN	replaced by frame	automatic
WCx	bathroom / toilet blocks	LOCKED IN	removed	automatic
BRx	balcony railings	LOCKED IN	removed	automatic



SPATIAL CONSEQUENCE

Complete removal of the external envelope. The interior space can extend outward, expanding the floor plan footprint through the integration of the new exoskeleton. The internal layout is flexible and can either be kept intact or completely transformed.

PROGRAMMATIC SUITABILITY

Suitable for enlarging existing apartments outward, or utilising the expanded floor plan space to create major open spaces for public and civic functions.

LEVEL OF INTERVENTION,COST AND CONSTRUCTION COMPLEXITY

Very high structural intervention. Maximum complexity and cost due to the facade removal and exoskeleton construction.

MATERIAL HARVEST

Very high (up to 75 %)



Figure 66. Scenario D spatial and programmatic evaluation

7.4 EXPERIMENT 3

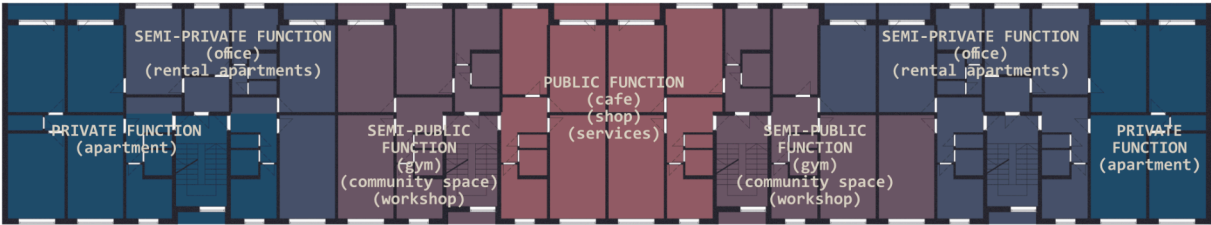
With the spatial and architectural consequences evaluated in Section 7.3, architectural intent could finally be introduced into experiment 3. It should, however, be noted that this test was not necessarily guided by the programmatic and functional injection determined in Section 2.2.4, but was rather a test of whether the analysis conducted in the previous section was correct.

The process of creating the schematic functional zoning within the plans was executed by categorising four levels of functions: public (shops, services), semi-public (community spaces, workshops), semi-private (rental apartments, offices), and private (apartments) aligning them directly with the four established scenarios. This approach was later met with the obstacle of not having a clear distinction between semi-public and semi-private spaces, as their functions seemed to inherently overlap. However, for the sake of this experiment, this zoning was maintained and directly translated into the grayscale image for the image sampler. In this direct translation, private functions were assigned Scenario A, semi-private functions - Scenario B, semi-public functions - Scenario C, and public functions - Scenario D.

This was also the first experiment where the higher-resolution image was used (see

Section 6.5) as the input for the sampler, offering greater variation within the shapes and assignments of the scenarios. Another novelty in this experiment was underlaying a floor plan in Affinity Designer to visualise and estimate which elements were being targeted in real-time, even during the functional diagram creation phase.

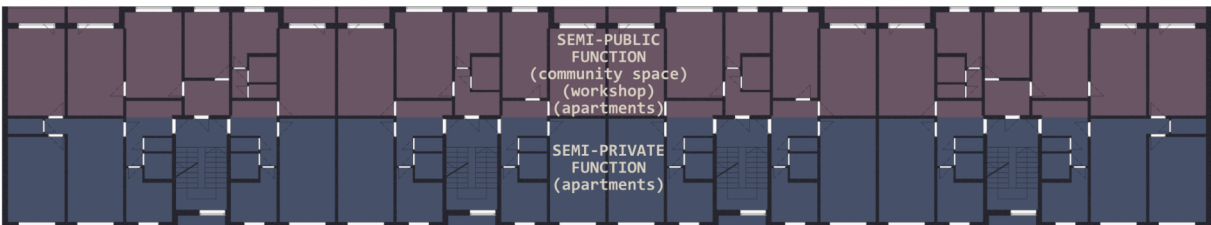
The outcomes of this experiment were conclusive: the dependency script works as intended; therefore, if Scenarios C or D are applied on the ground floor, they are automatically applied on all floors above, and the original image for the image sampler is overwritten. This consequently proves the issue discussed in Section 5.3 regarding the architectural need versus structural reality, but in this case, specifically concerning layout variability possibilities. If a public function requiring a large open space – and therefore needing a Scenario C or D application – is injected into the ground floor, the consequence is these same scenarios being applied throughout the building. This discovery reveals two significant aspects: first, the tool has to be used with precise intention, and the removal of walls in large areas must be strictly evaluated as the best solution. Second, the user of the tool must remain flexible regarding the planned versus real outputs, being able to adjust the planned programmatic injection to the generated structural output.



GROUND FLOOR PLAN



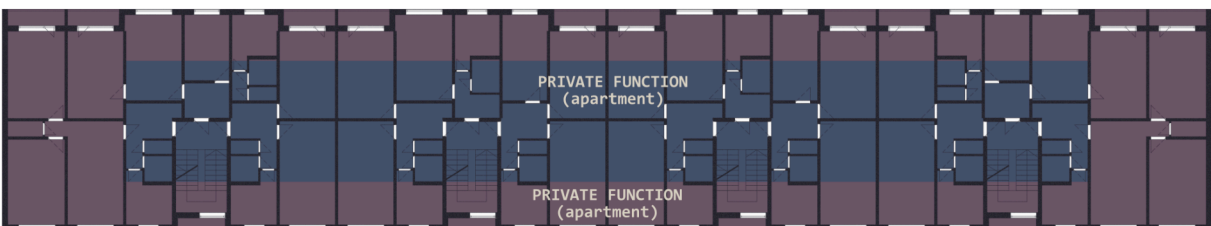
2ND FLOOR PLAN



3RD FLOOR PLAN



4TH FLOOR PLAN



5TH FLOOR PLAN

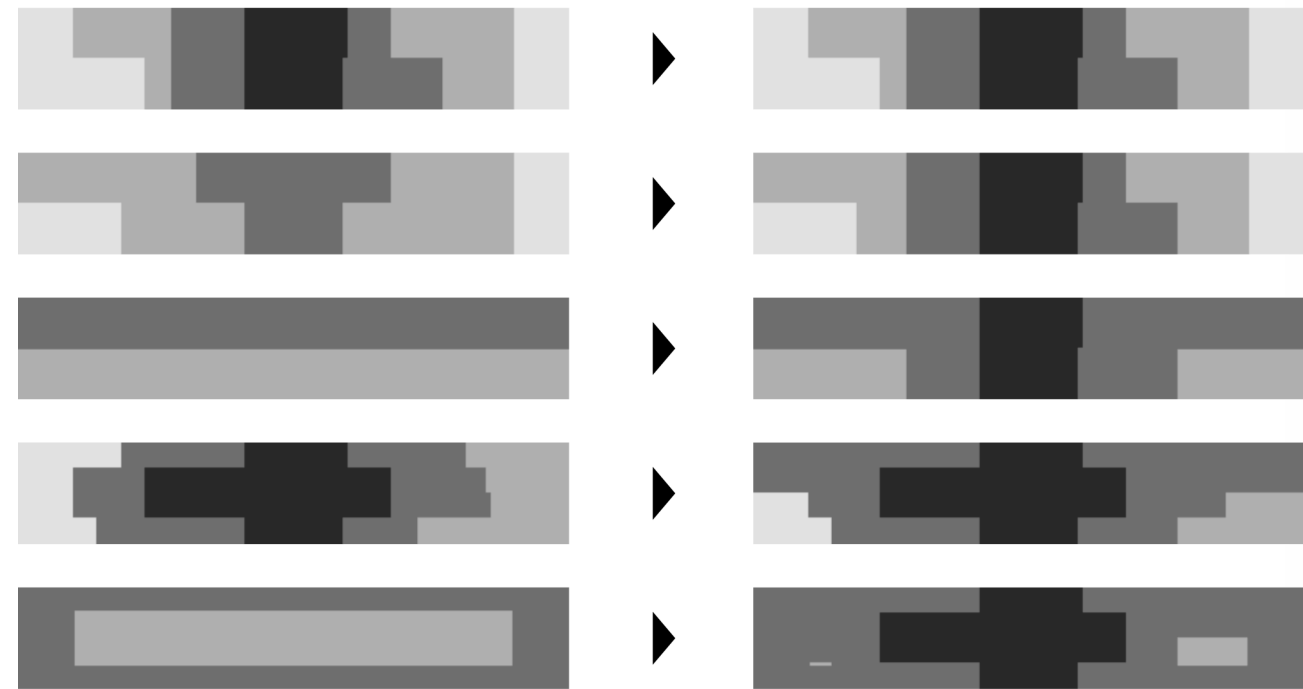
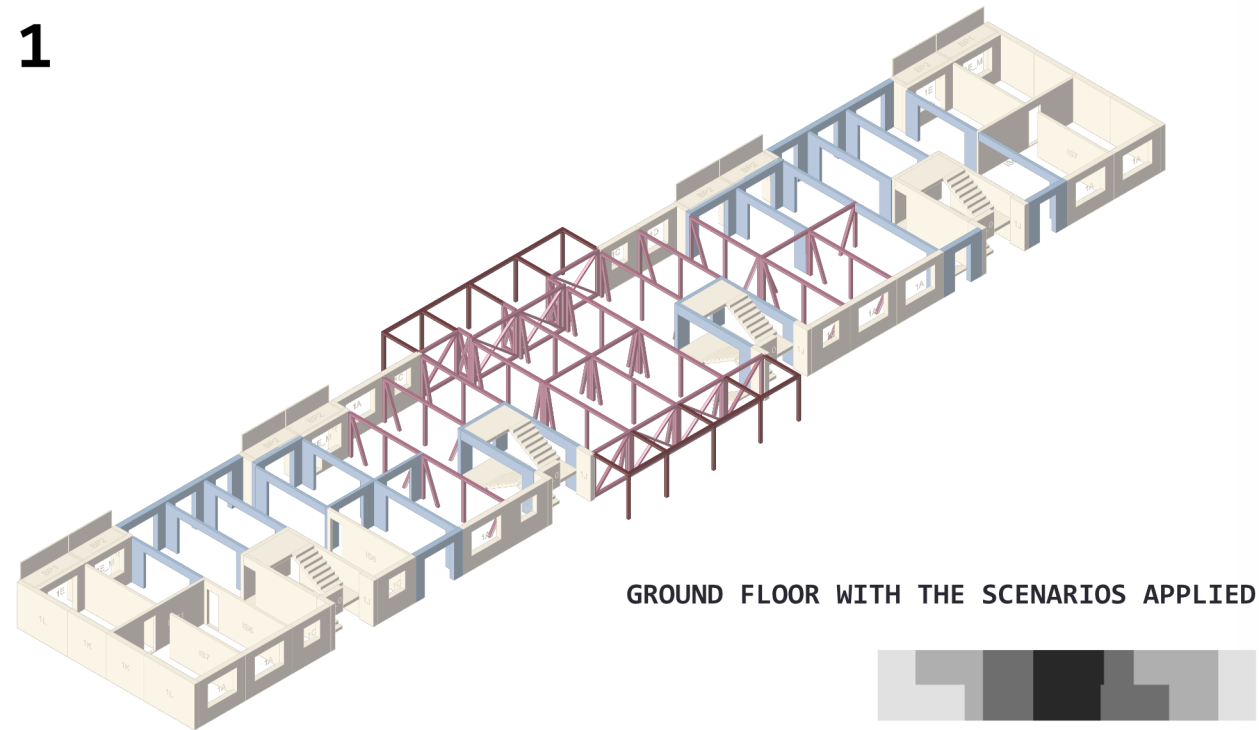


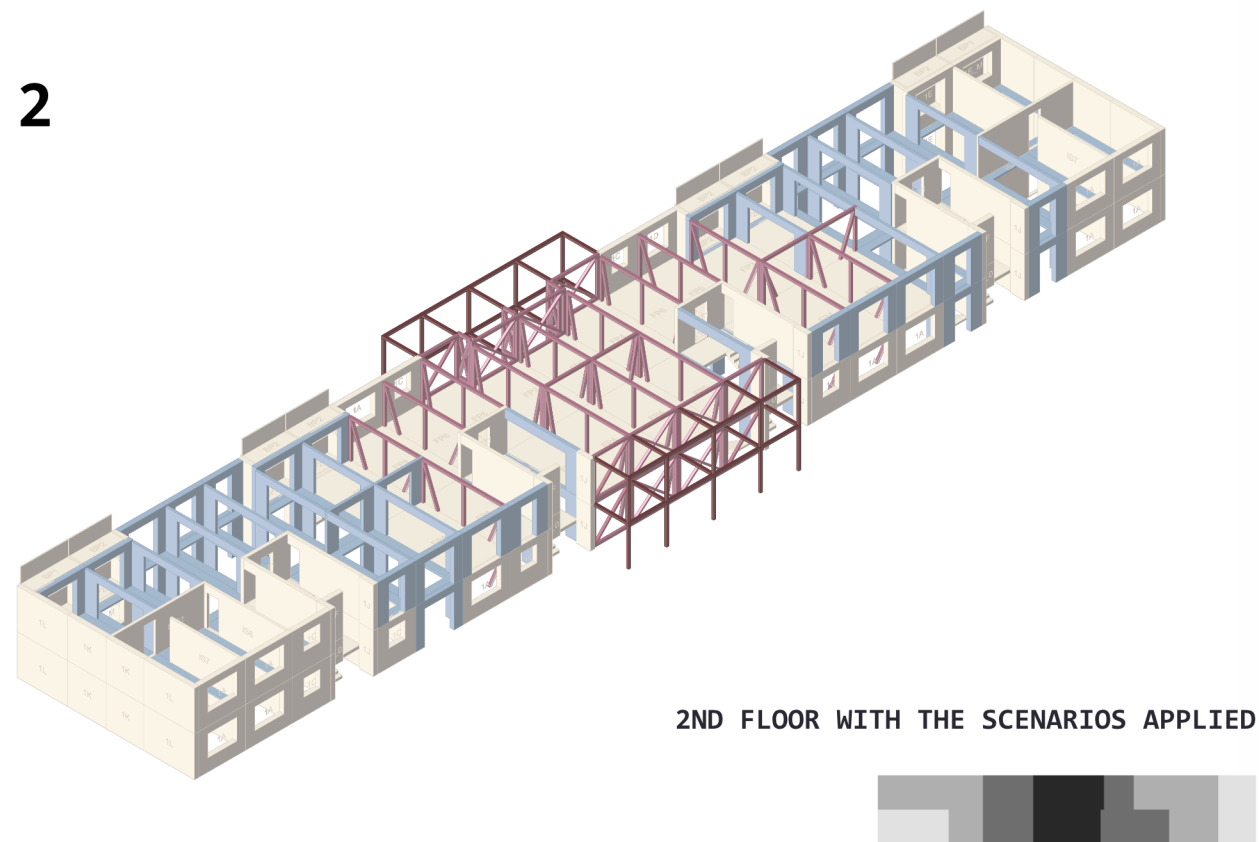
Figure 67. The original inputs of the image sampler (left) and the images adjusted through the application of forced scenarios (right) for Experiment 3

Figure 68. Programmatic diagrams of all the floors of the building for Experiment 3

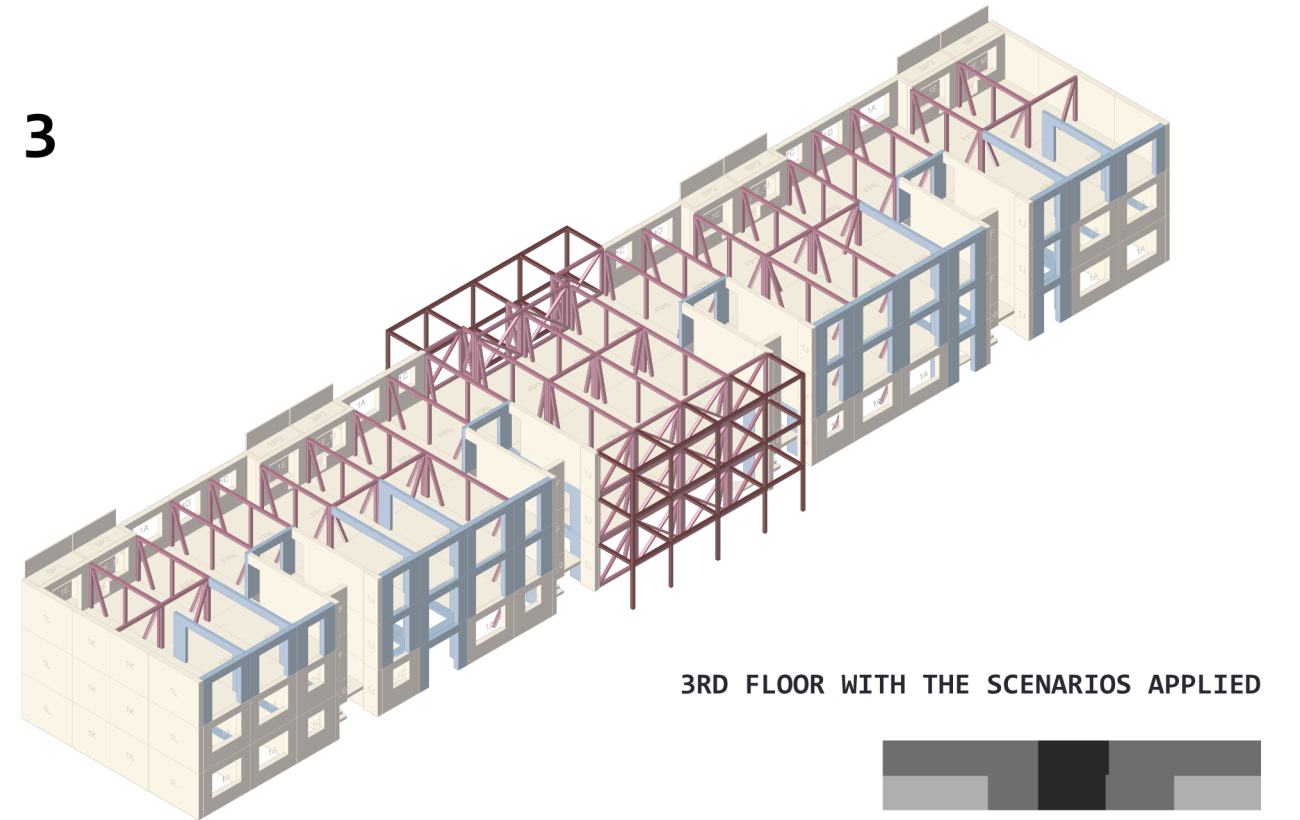
1



2



3



4

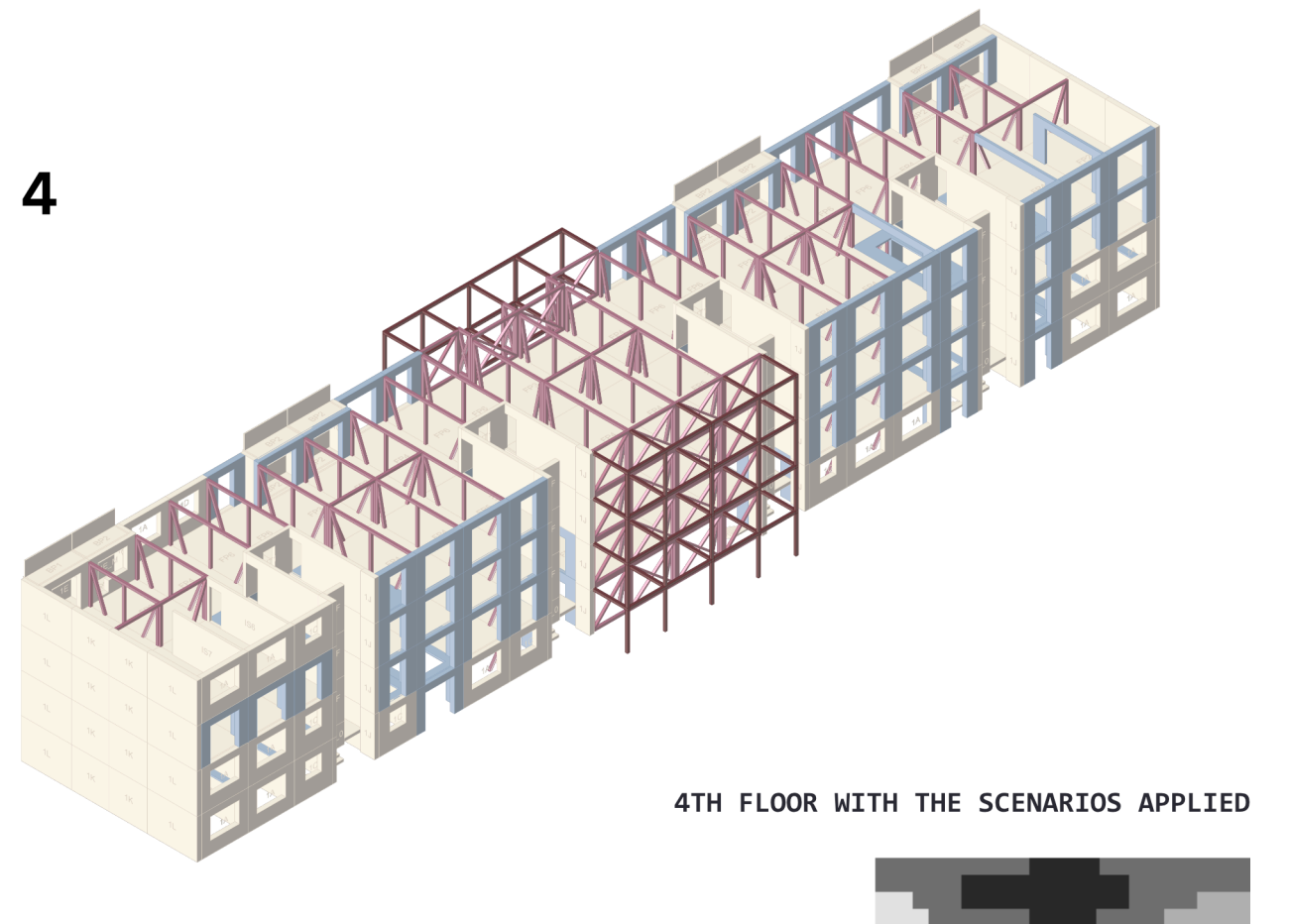
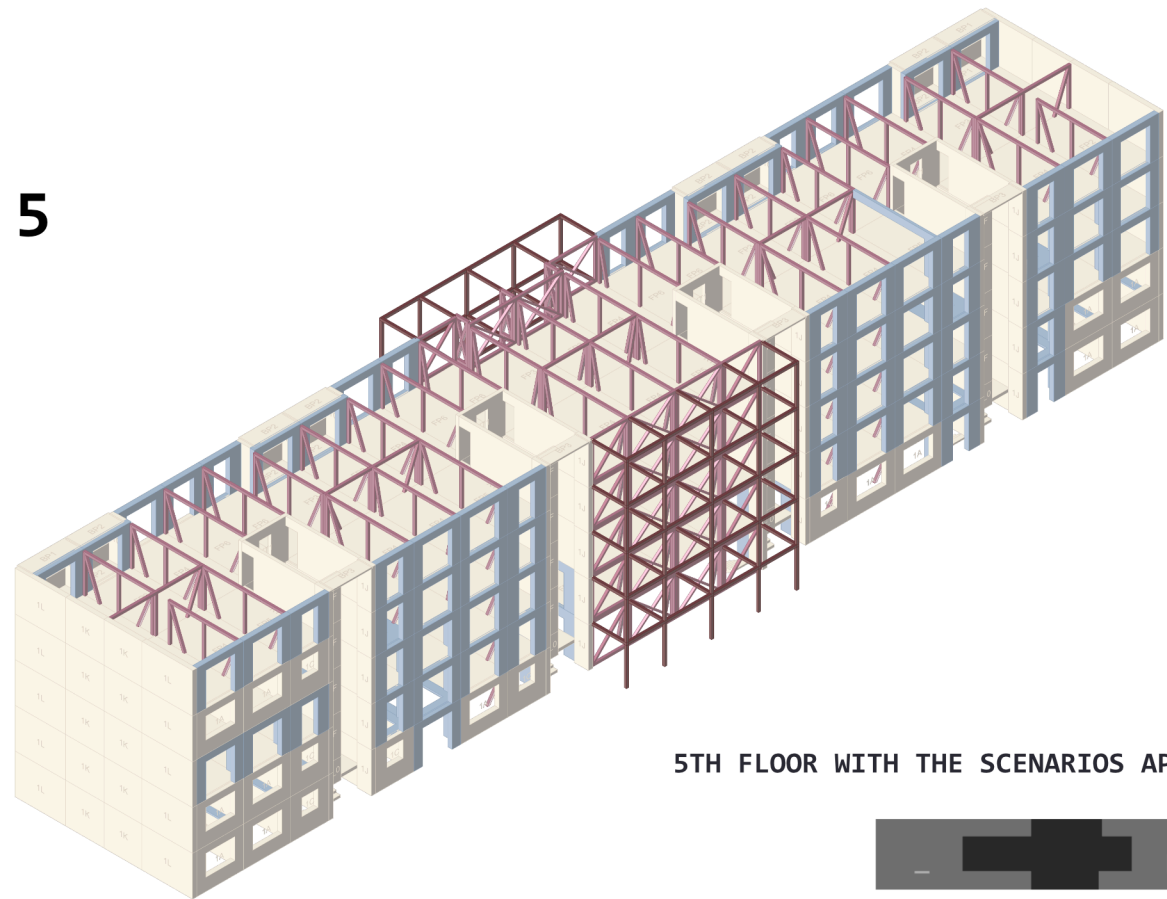


Figure 69. Script generated geometries for floors 1-4 in Experiment 3

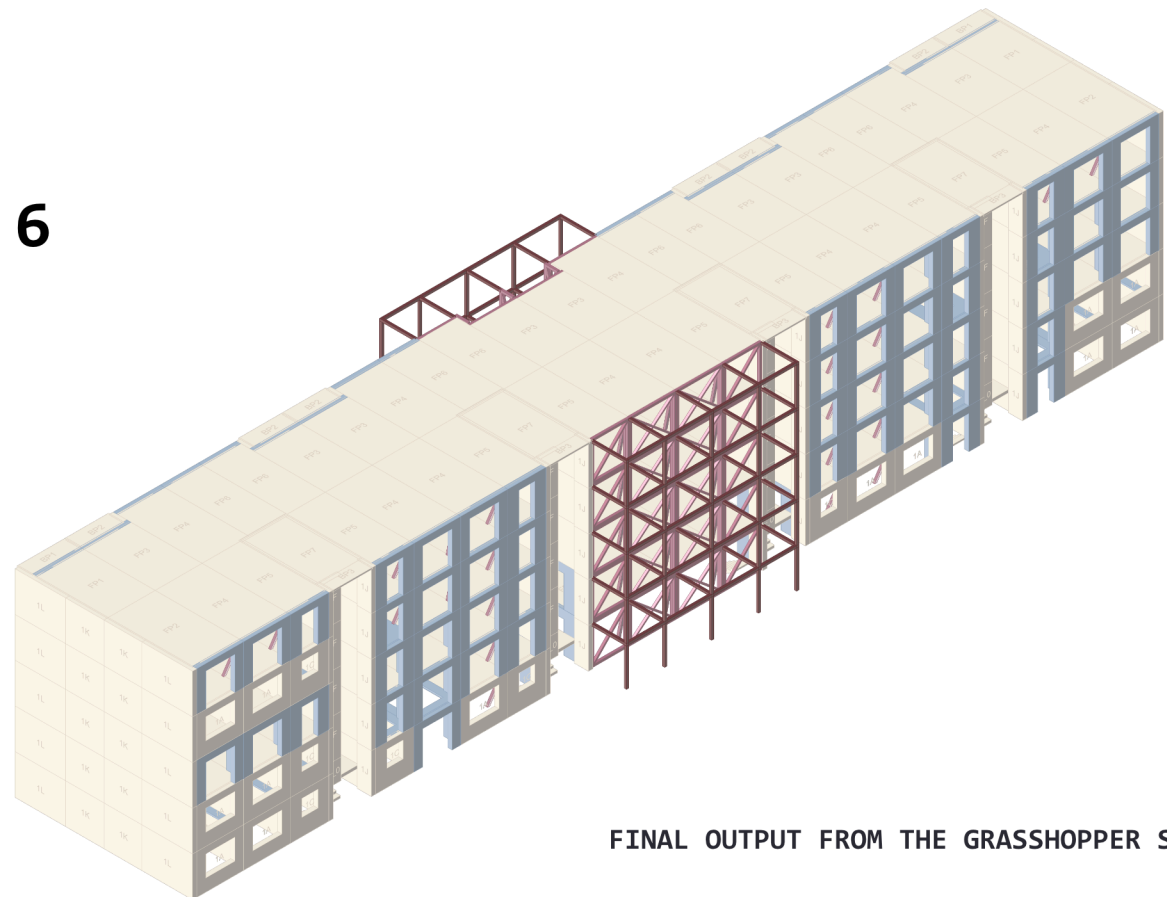
5



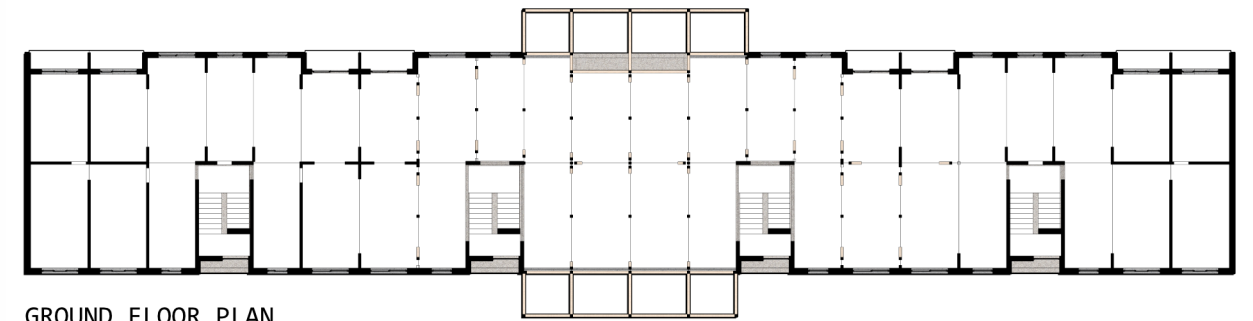
5TH FLOOR WITH THE SCENARIOS APPLIED



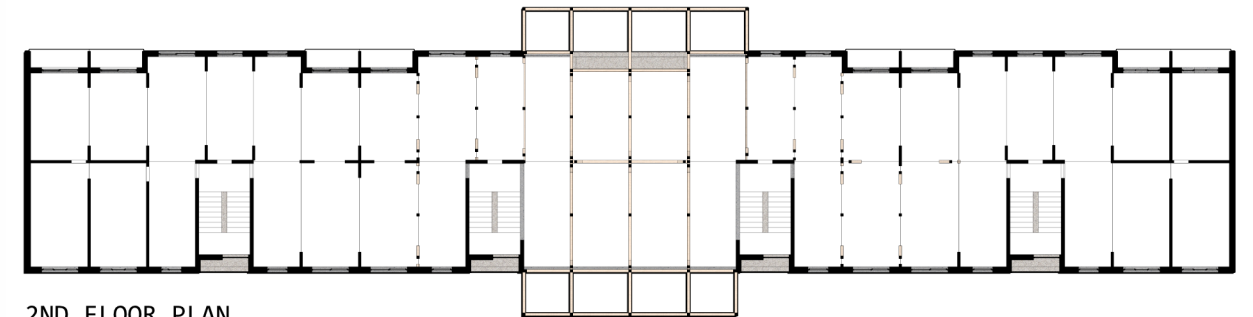
6



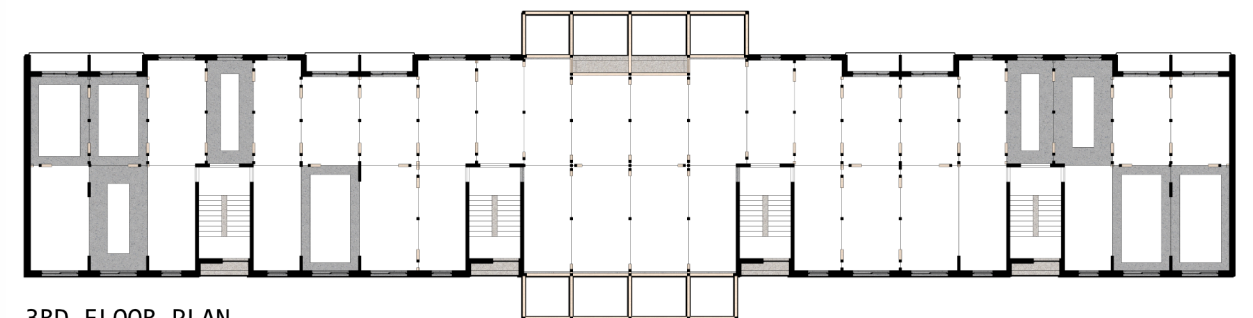
FINAL OUTPUT FROM THE GRASSHOPPER SCRIPT



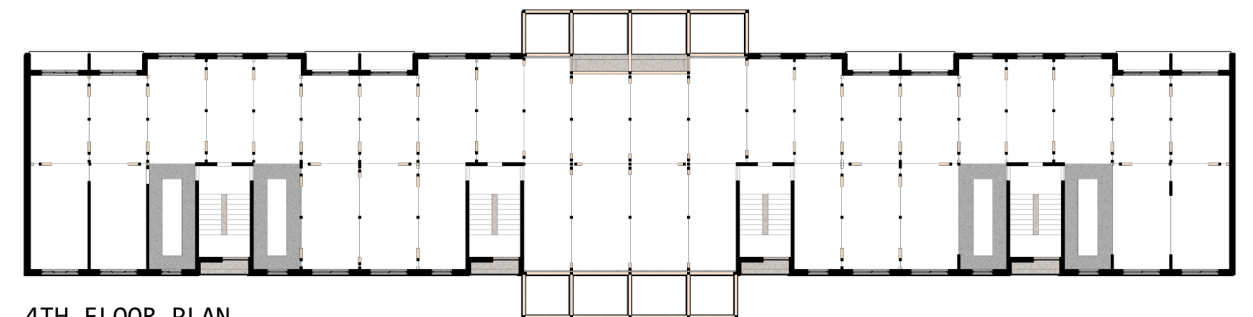
GROUND FLOOR PLAN



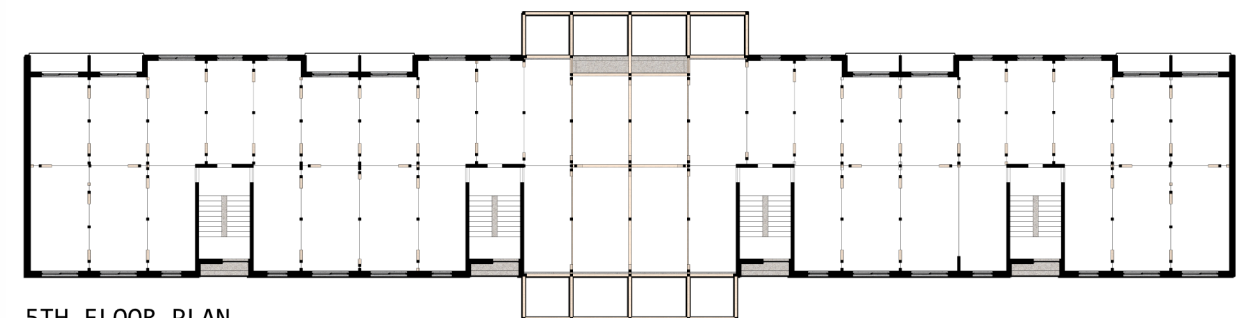
2ND FLOOR PLAN



3RD FLOOR PLAN



4TH FLOOR PLAN



5TH FLOOR PLAN

Figure 70. Script generated geometries for floor 5 and final script output geometry in Experiment 3

Figure 71. Floor plan outputs from the script generated geometry

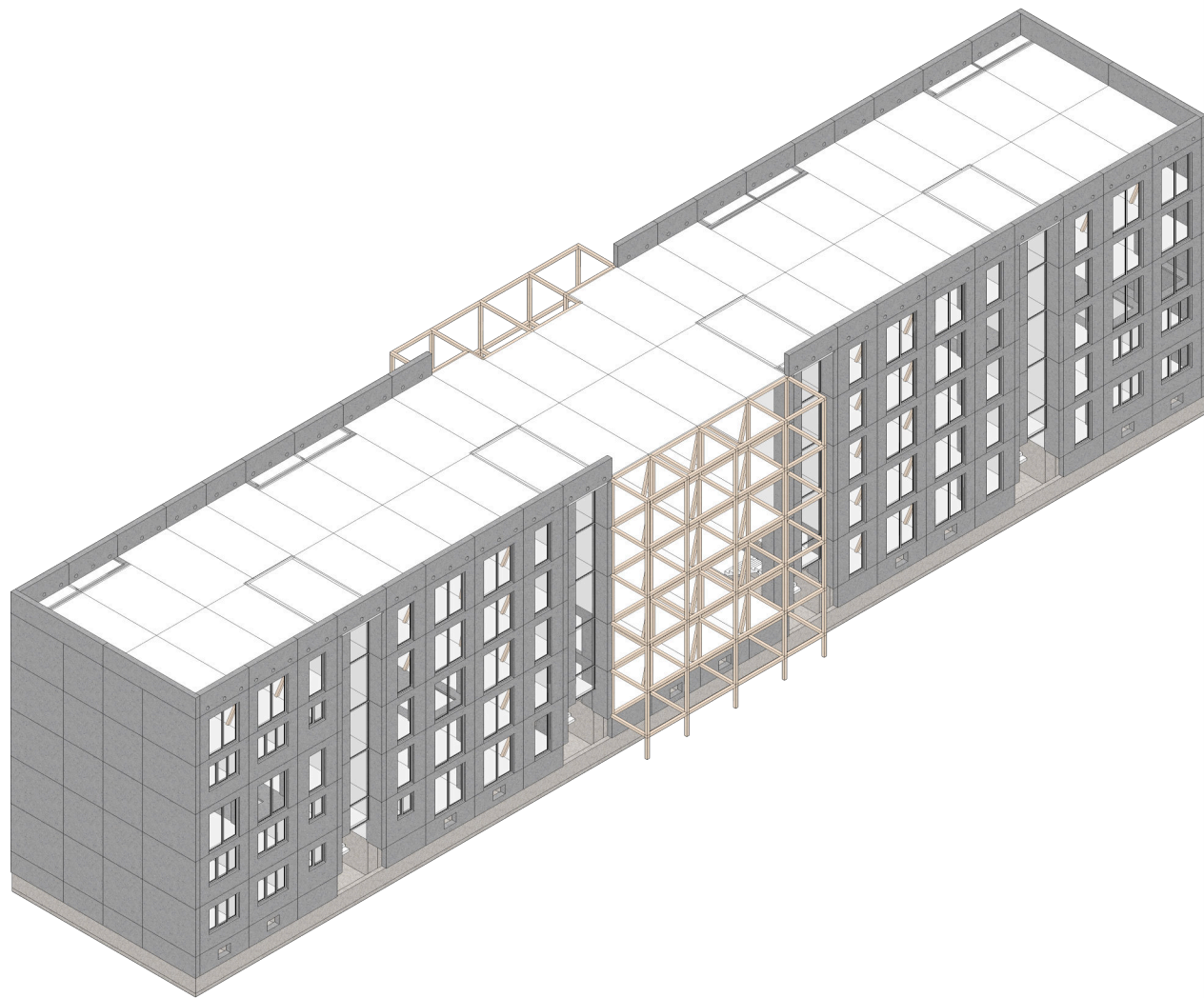


Figure 72. An example of the post-algorithmic alteration output for Experiment 3

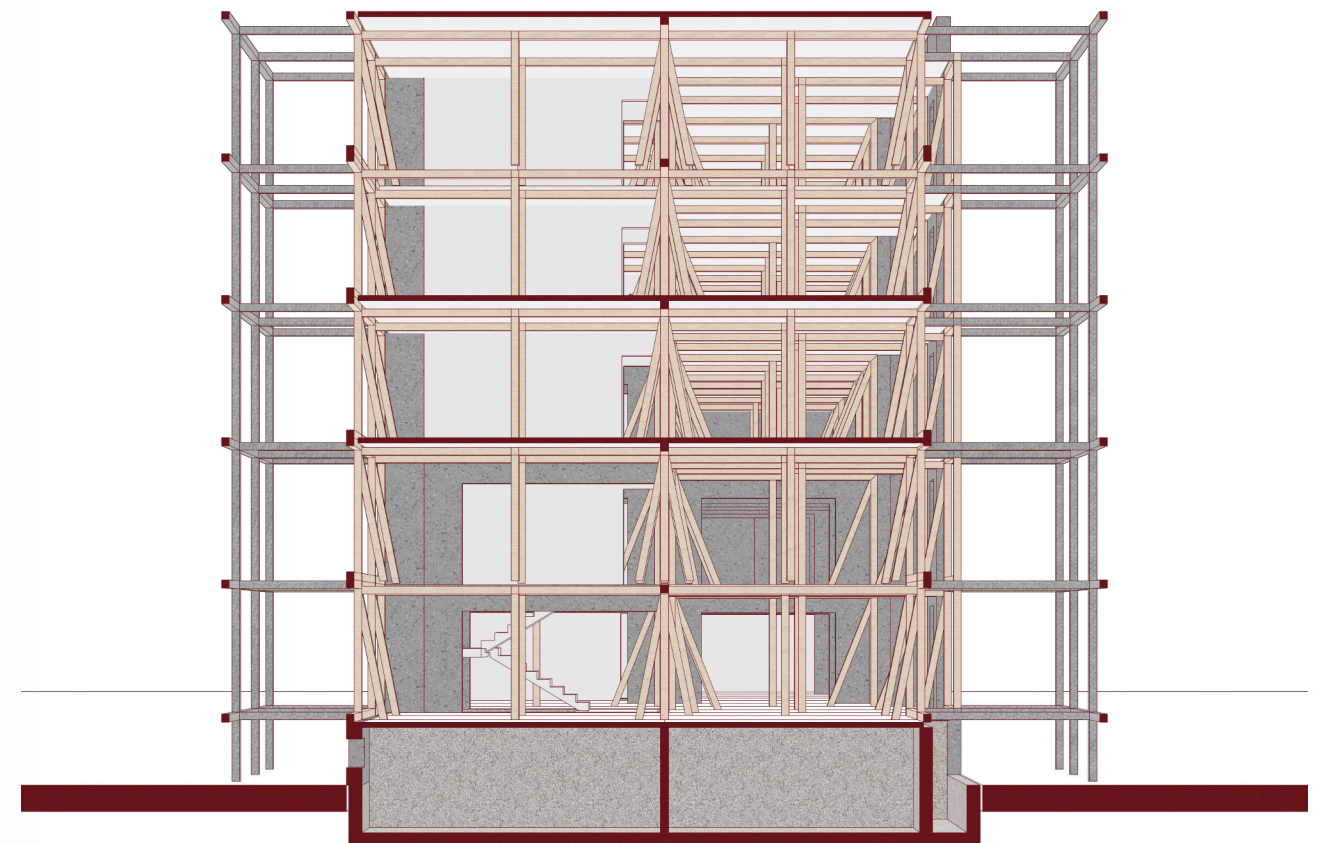


Figure 73. A section of the altered geometry output from Experiment 3

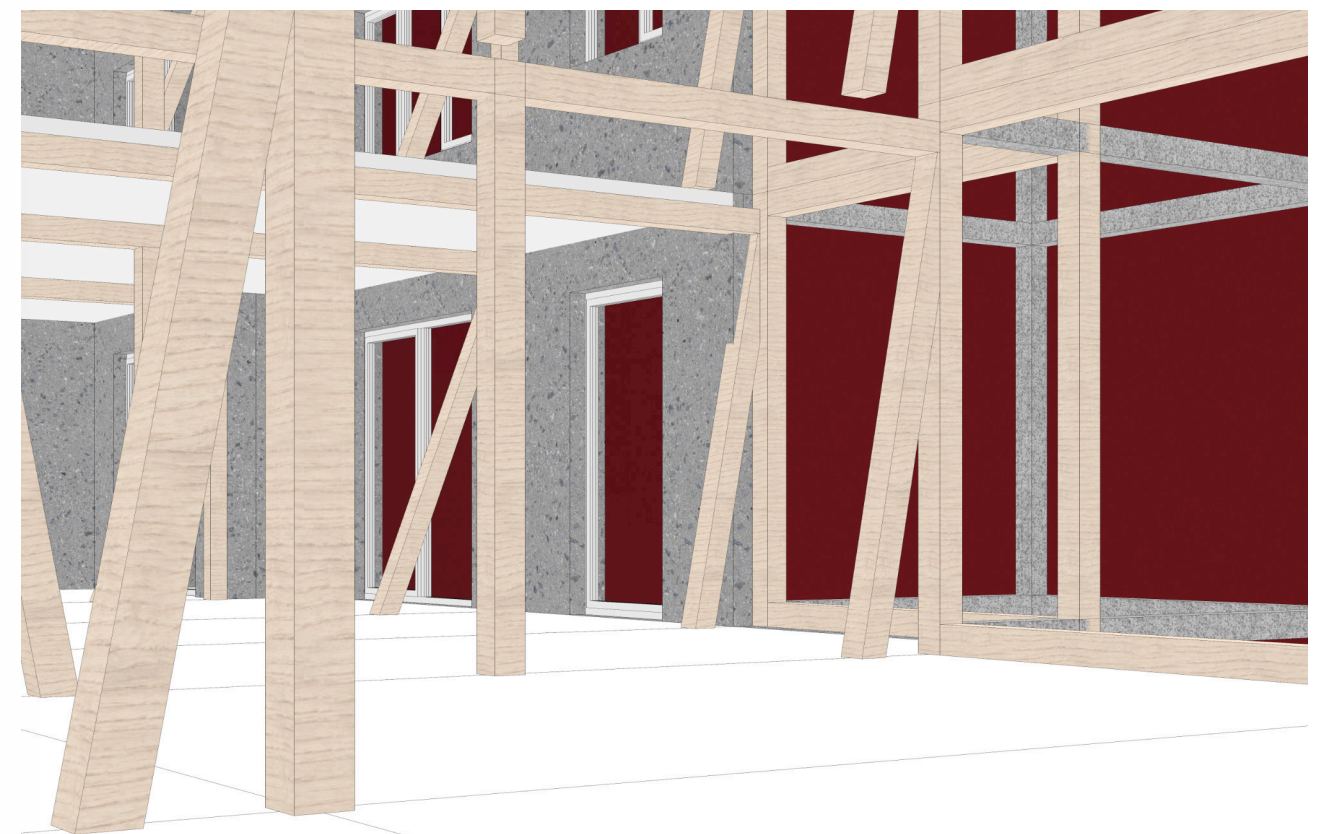


Figure 74. An example of the new-developed interior space from Experiment 3

7.5 EXPERIMENT 4

With the limitations of the tool identified in Experiment 3, a new approach was tested in Experiment 4. The functional diagrams were created with clear architectural intention, taking into account the programmatic injections defined in Section 2.2.4 – public and community spaces. For this experiment, the previous approach of defining four broad space categories was discarded, as it did not produce the desired results. Instead, the functional diagrams were created with specific functions, visible in Figure 76. Another novelty in this iteration was the predefinition of spaces that needed to span two storeys vertically.

As it was previously established that the application of specific scenarios must be carefully targeted towards specific elements rather than broad spatial “blocks”, the functional diagrams were underlaid in the grayscale image creation phase instead of plain floor plans. This allowed the user to apply certain scenarios to targeted elements, rather than uniformly across the entire space. Consequently, the grayscale images for this experiment are visibly more fragmented, as observed in Figure 75. For example, the walls between distinct functions were assigned Scenario A to ensure minimal intervention and a preserved physical boundary between them. The application of Scenarios C and D was carefully evaluated and executed only on

targeted walls in the lower floors to prevent a complete “replacement” override.

A further advancement in this test is the assignment of floor slab scenarios – within this iteration, the script was supplemented with a separate image sampler specifically for the floor slab panels. Previously, the system was flawed, as it did not allow for targeting specific floor slabs, but because this test introduced the specifically targeted multi-story space allocation, this step was necessary to ensure maximum user-guided control over the output.

The conclusions drawn from this experiment confirm that the “fragmented” approach of assigning scenarios to specific elements rather than whole spaces is the optimal method, as it provides significantly better control over the process. The introduction of the floor slab image sampler was also evaluated as a meaningful addition, providing additional spatial control over the vertical dimension of spaces. However, this test identified another imperfection: as shown in Figure 76, the “larger” functions, such as apartments, were still defined within the functional diagram as singular large blocks. In order to prevent these volumes from merging into oversized apartments, the outlines of the units desired must be defined within these diagrams. This adjustment would allow the user to gain maximal control over the final output.

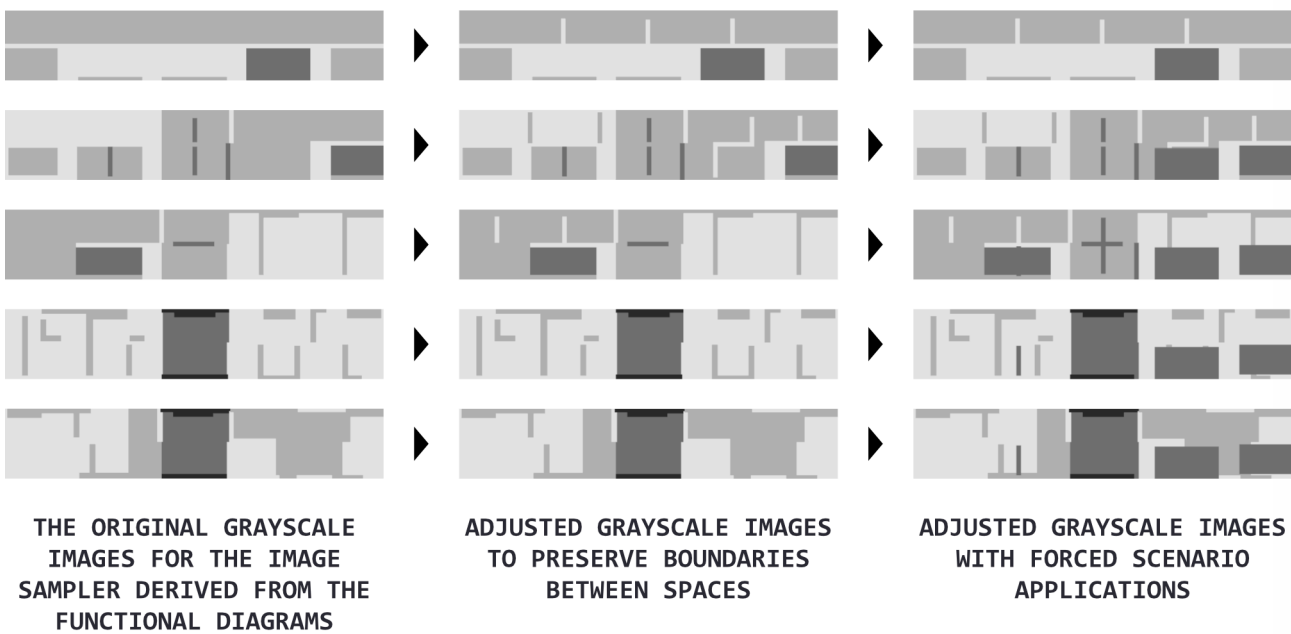


Figure 75. The original inputs of the image sampler (left), the manually modified inputs (middle) and the images adjusted through the application of forced scenarios (right) for Experiment 4



GROUND FLOOR PLAN



2ND FLOOR PLAN



3RD FLOOR PLAN



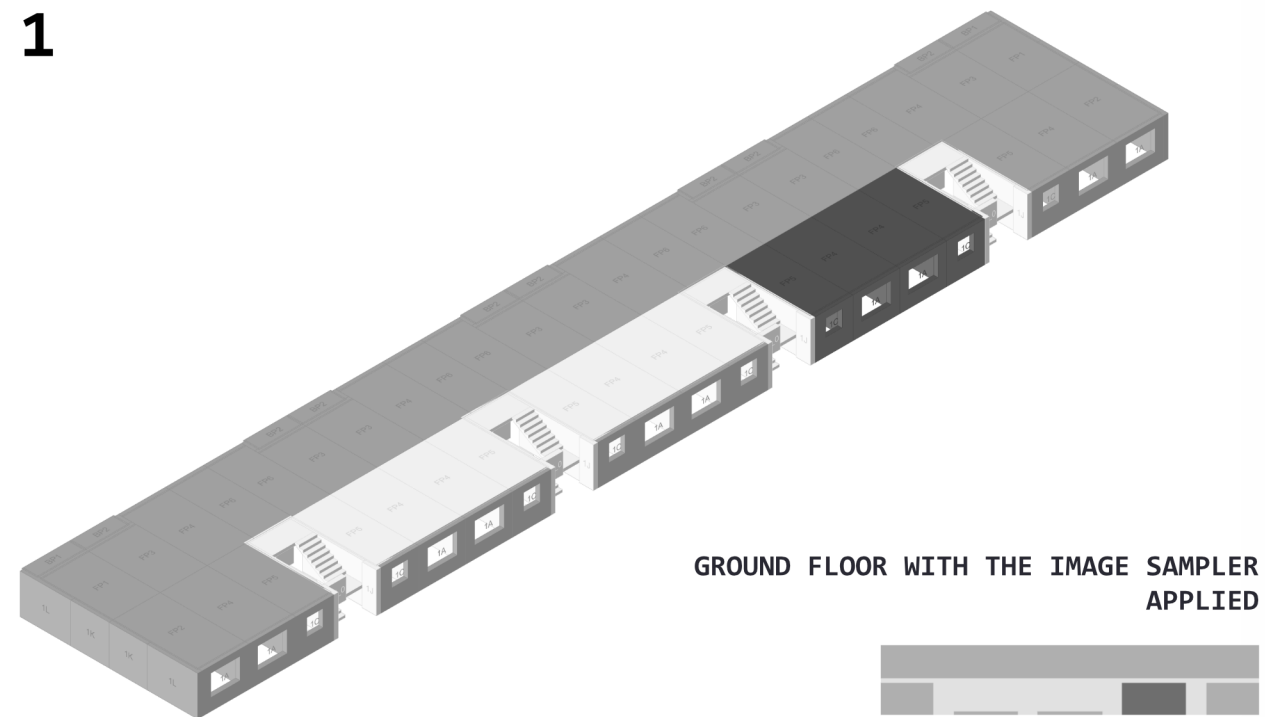
4TH FLOOR PLAN



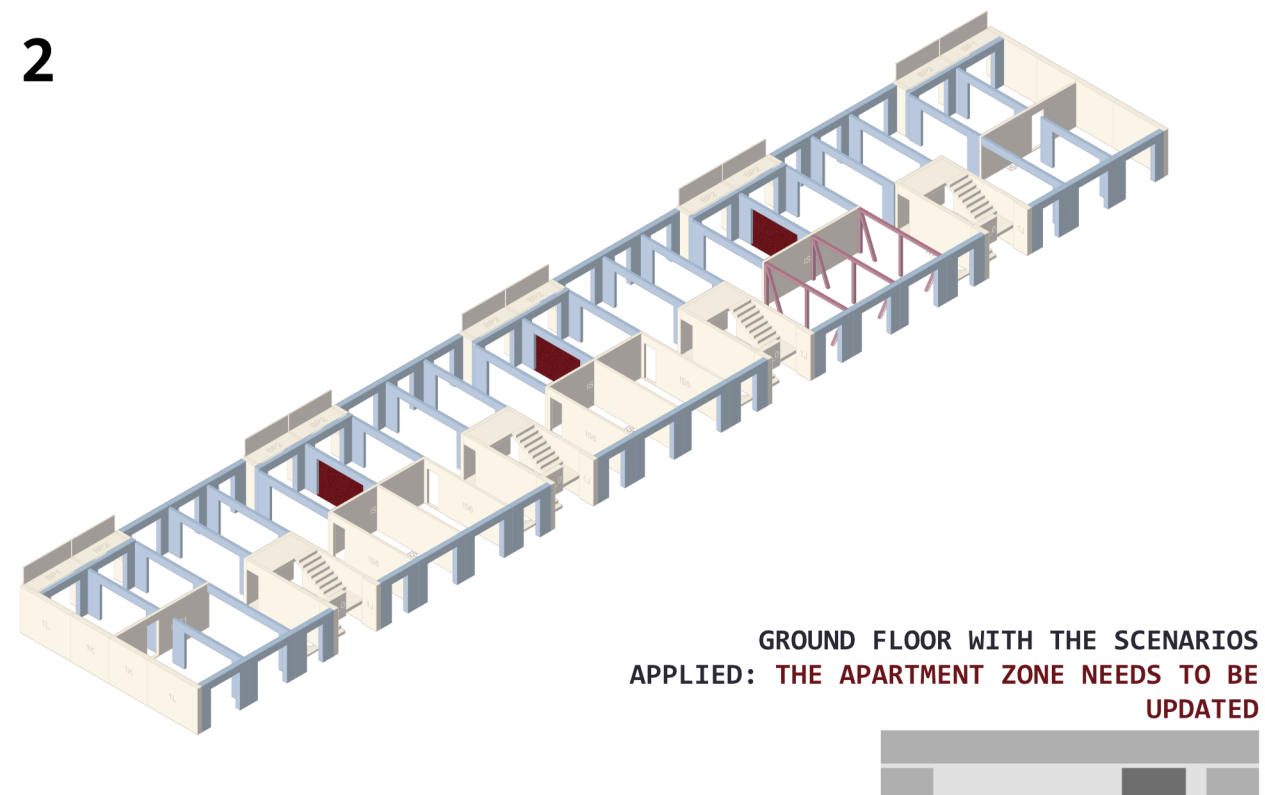
5TH FLOOR PLAN

Figure 76. Programmatic diagrams of all the floors of the building for Experiment 4

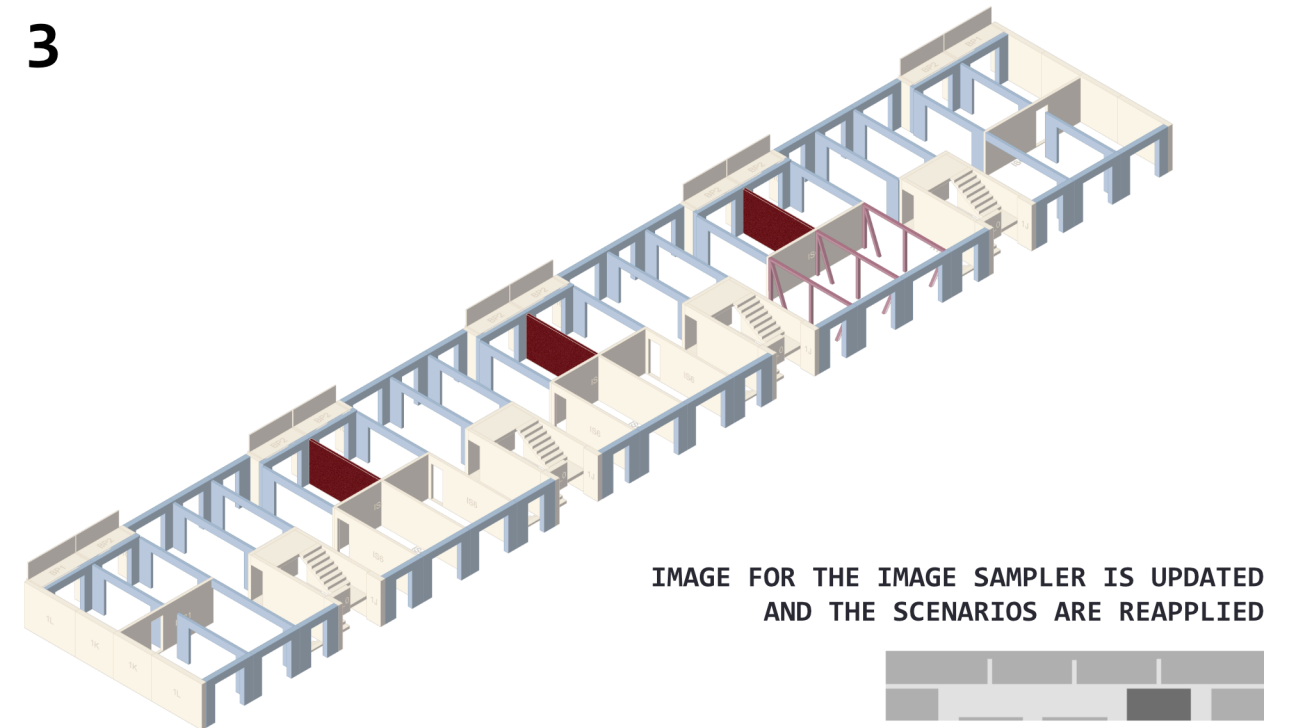
1



2



3



4

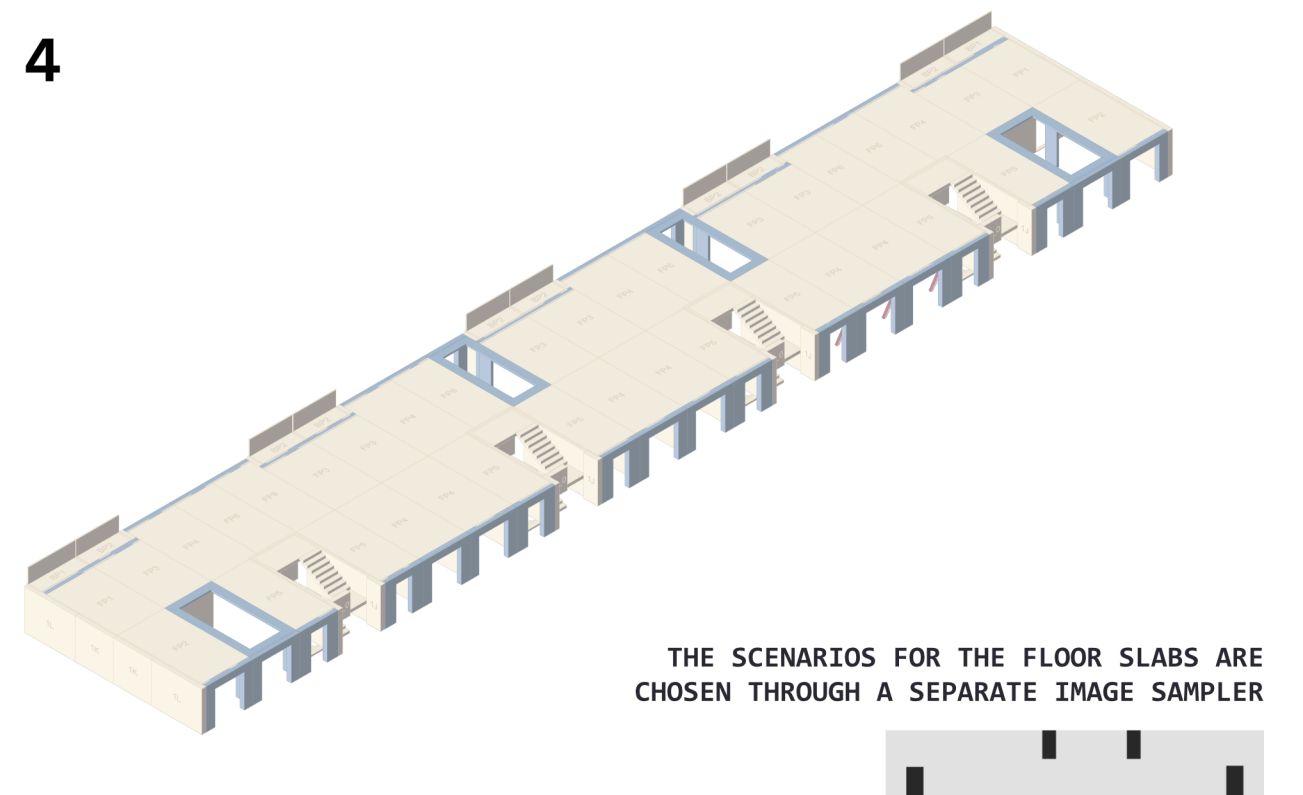
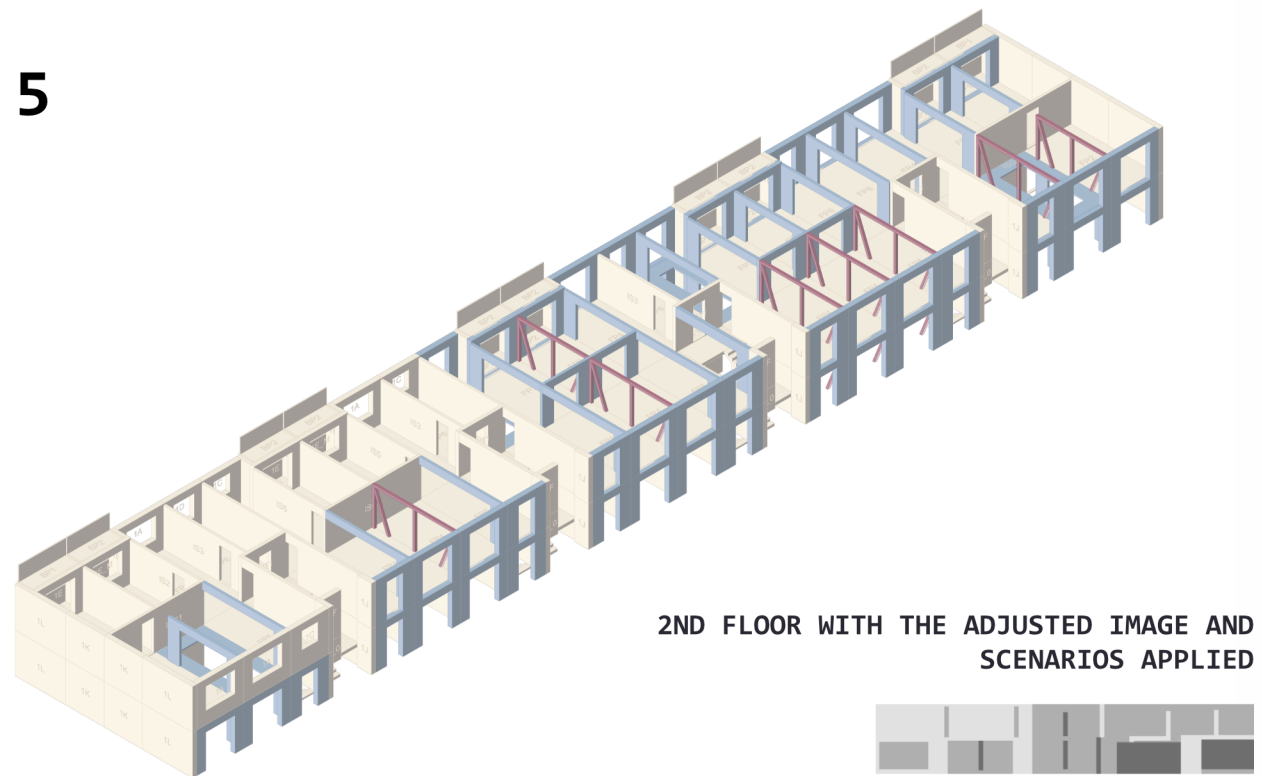


Figure 77. Script application trials and iterations for the ground floor in Experiment 4

5



6

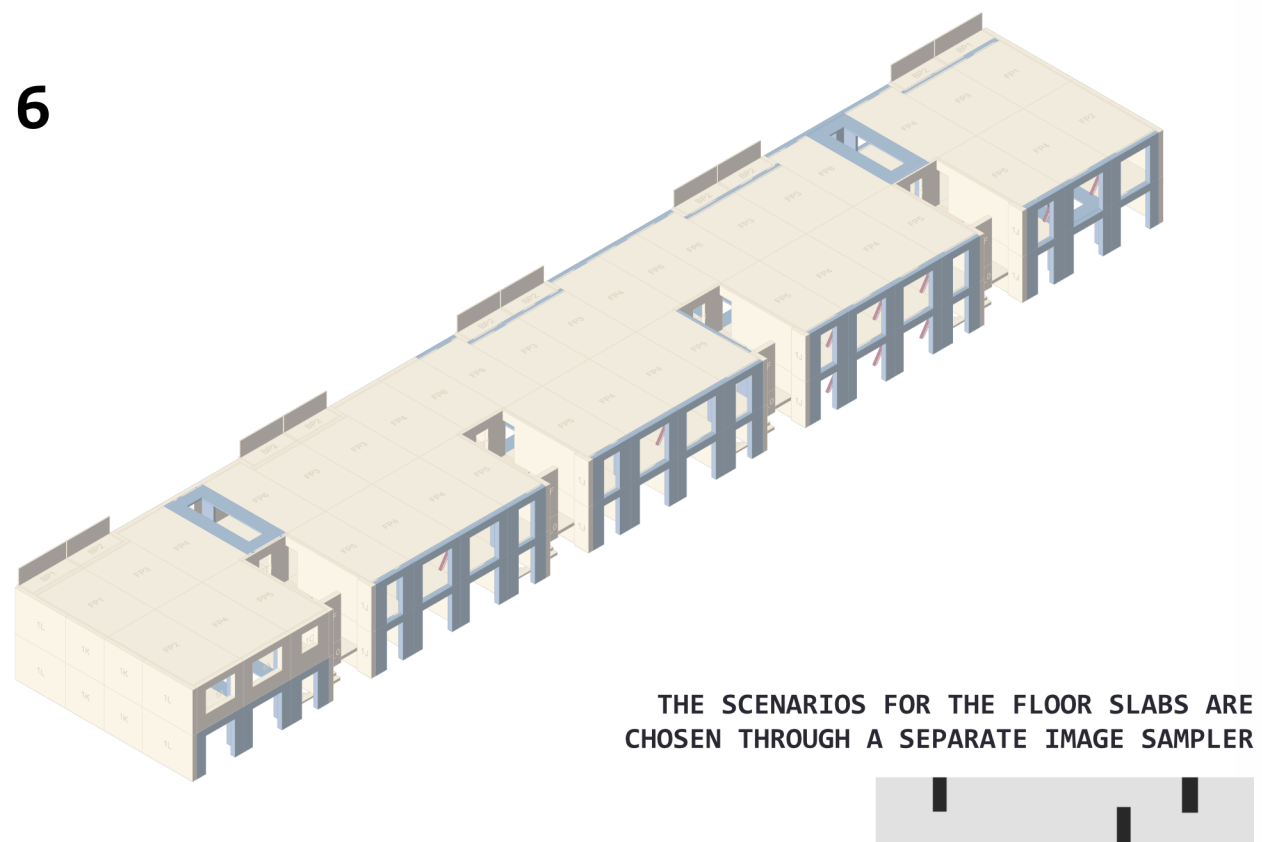


Figure 78. Script application for the 2nd floor in Experiment 4

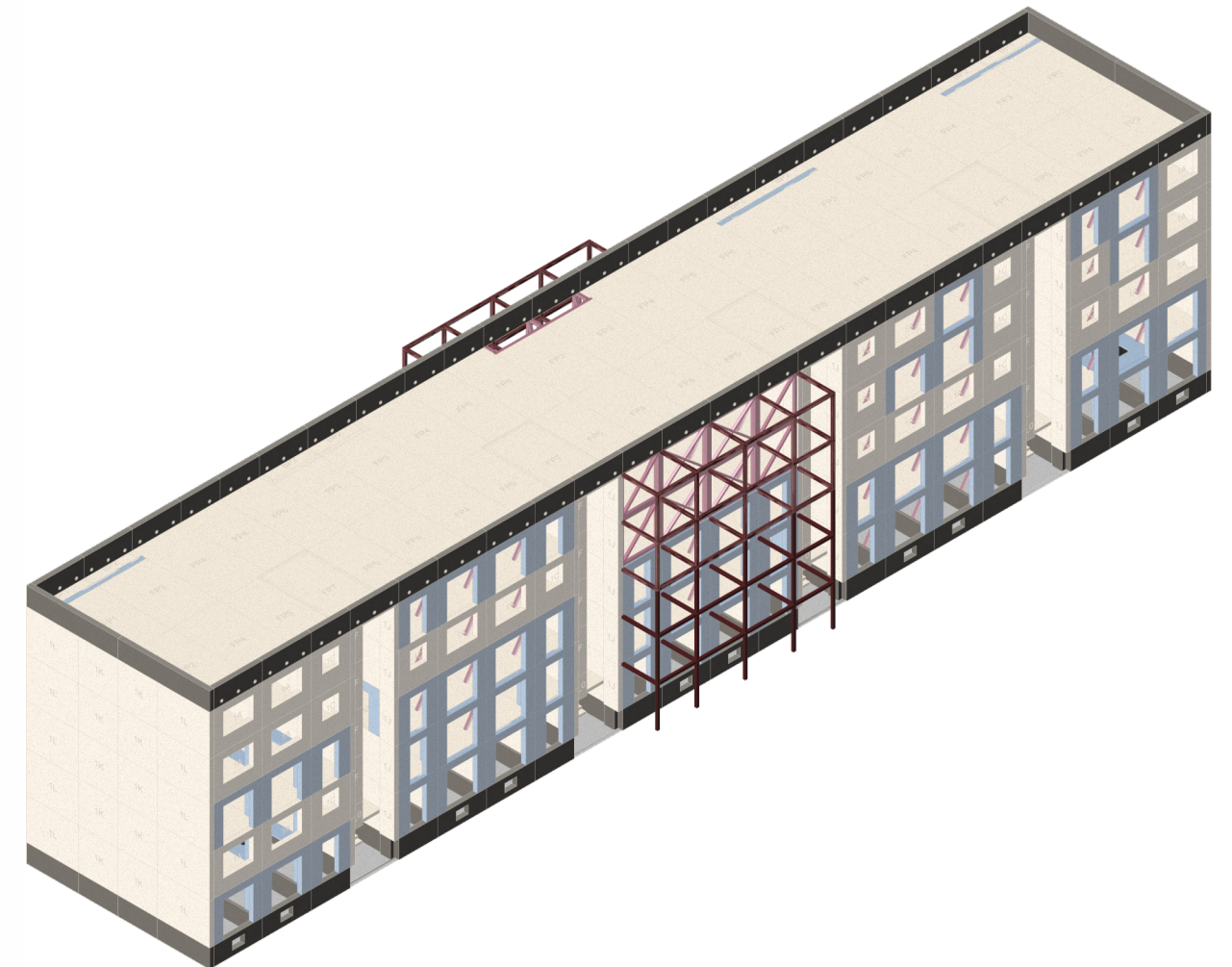


Figure 79. Output geometry from the algorithm with minor manual adjustments in Experiment 4

7.6 EXPERIMENT 5

Experiment 5 was the culminating test and the one that resulted in the final architectural synthesis. It was conducted based on the insights gathered from previous trials and the resulting improvements executed on the algorithmic tool. This concluding iteration serves as proof that the tool works as intended, but that the results highly depend on the user in both the pre-algorithmic and post-algorithmic steps, therefore positioning the developed system as exactly that – a tool for the architect, and not vice versa.

The first step of this phase, as in previous iterations, was the creation of detailed functional diagrams. The findings from the previous experiment dictated that the preferred outlines of each specific functional space should be defined within this step; therefore, the large blocks of apartments were split into anticipated future unit configurations, taking into account the existing plumbing nodes and connections to staircases (Figure 84). Other functions were evaluated individually, for example, a gym space was not split into smaller segments as it requires open floor space.

The next step was again translating these functional floor plan diagrams into grayscale images. In this trial, the intermediate step of adjusting the image sampler inputs after running the script was omitted; instead, the behaviour of the script was anticipated based on the previous experience of the user.

The process was run successfully through all building floors, and a desired output was achieved. As in all prior tests, the leftover concrete and removed elements were accounted for to reserve them for reuse in post-algorithmic steps (Figure 85). Once the algorithmic step for all floors was executed, the newly established floor plans were extracted and compared against the original ones to adjust the functional diagram (Figure 84). During this extraction, places where new elements were needed were identified to “flag” them as positions for reclaimed element insertion. With that, Experiment 5 was concluded and the architectural synthesis step could begin.

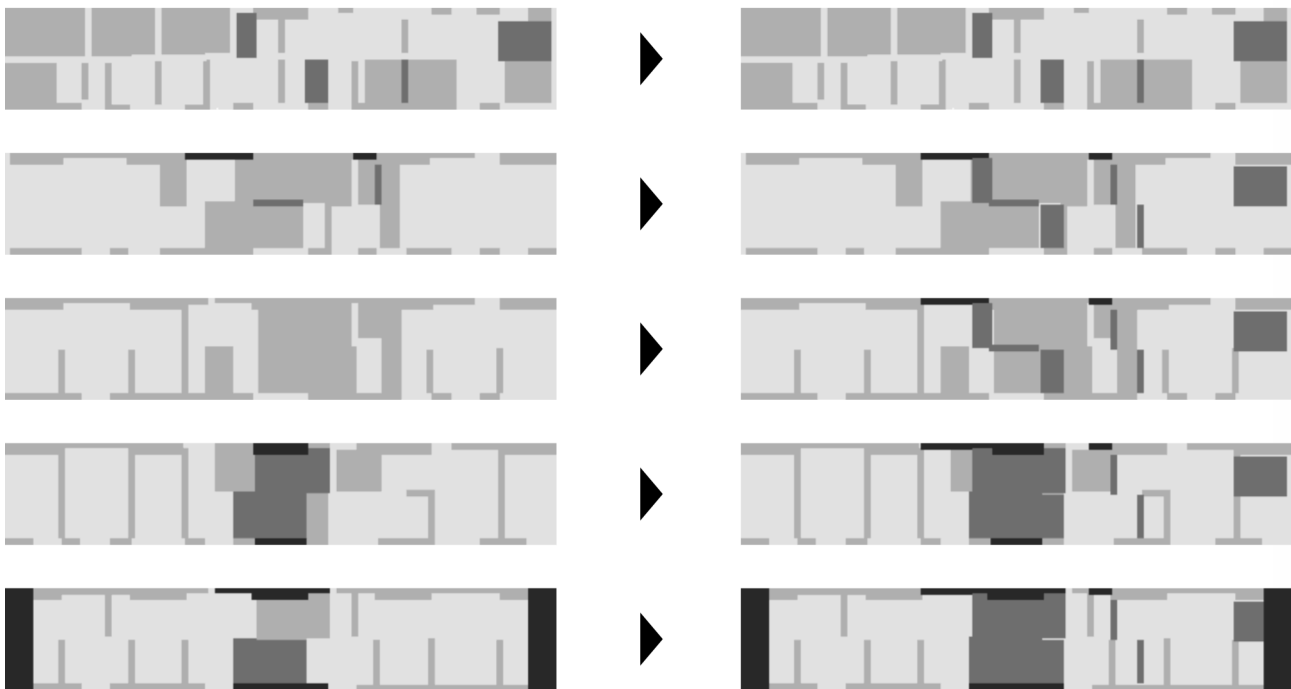


Figure 80. The original inputs of the image sampler (left) and the images adjusted through the application of forced scenarios (right) for Experiment 5

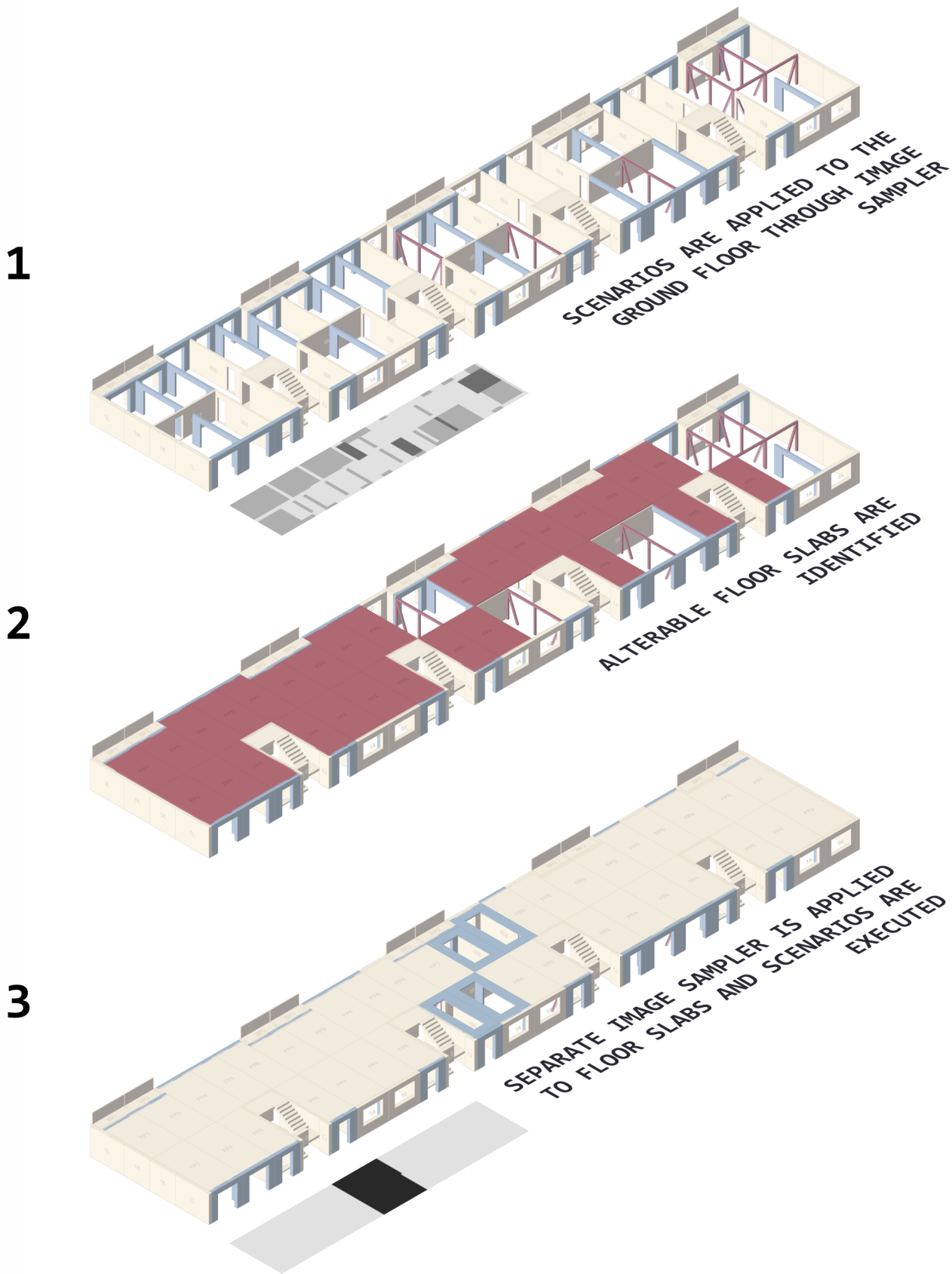


Figure 81. Algorithm application on the ground floor elements for Experiment 5

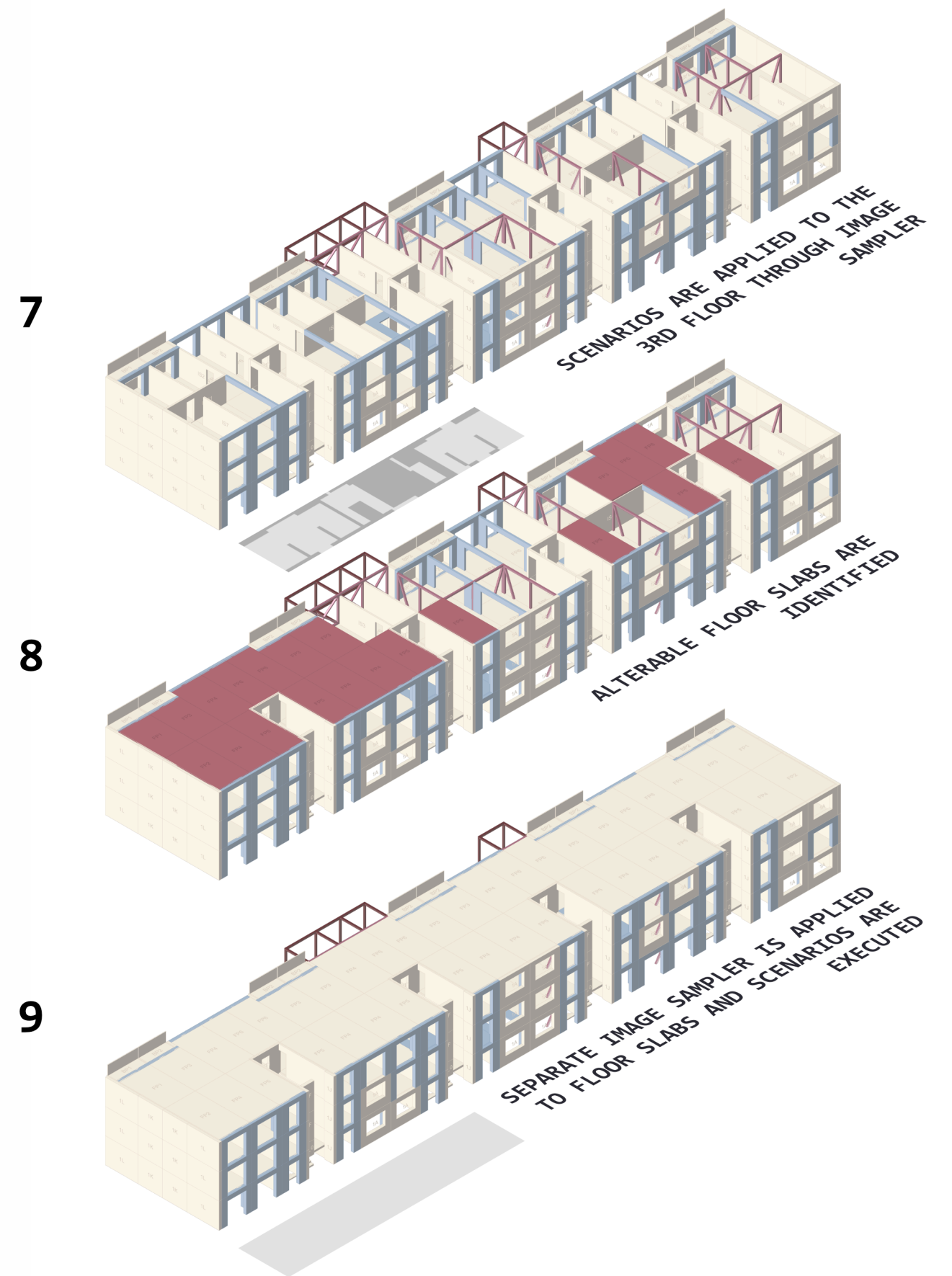
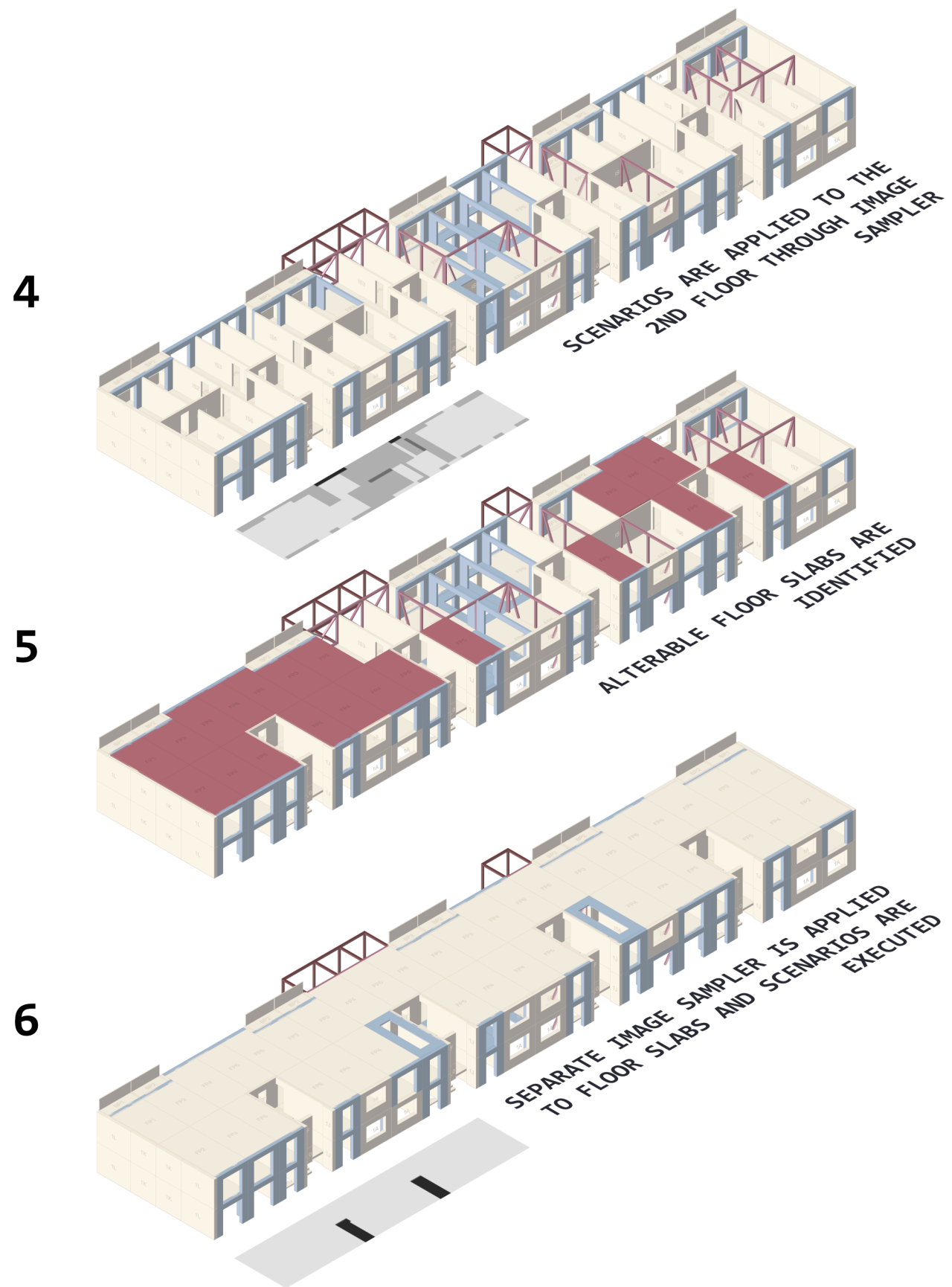


Figure 82. Algorithm application on the 2nd and 3rd floor elements for Experiment 5

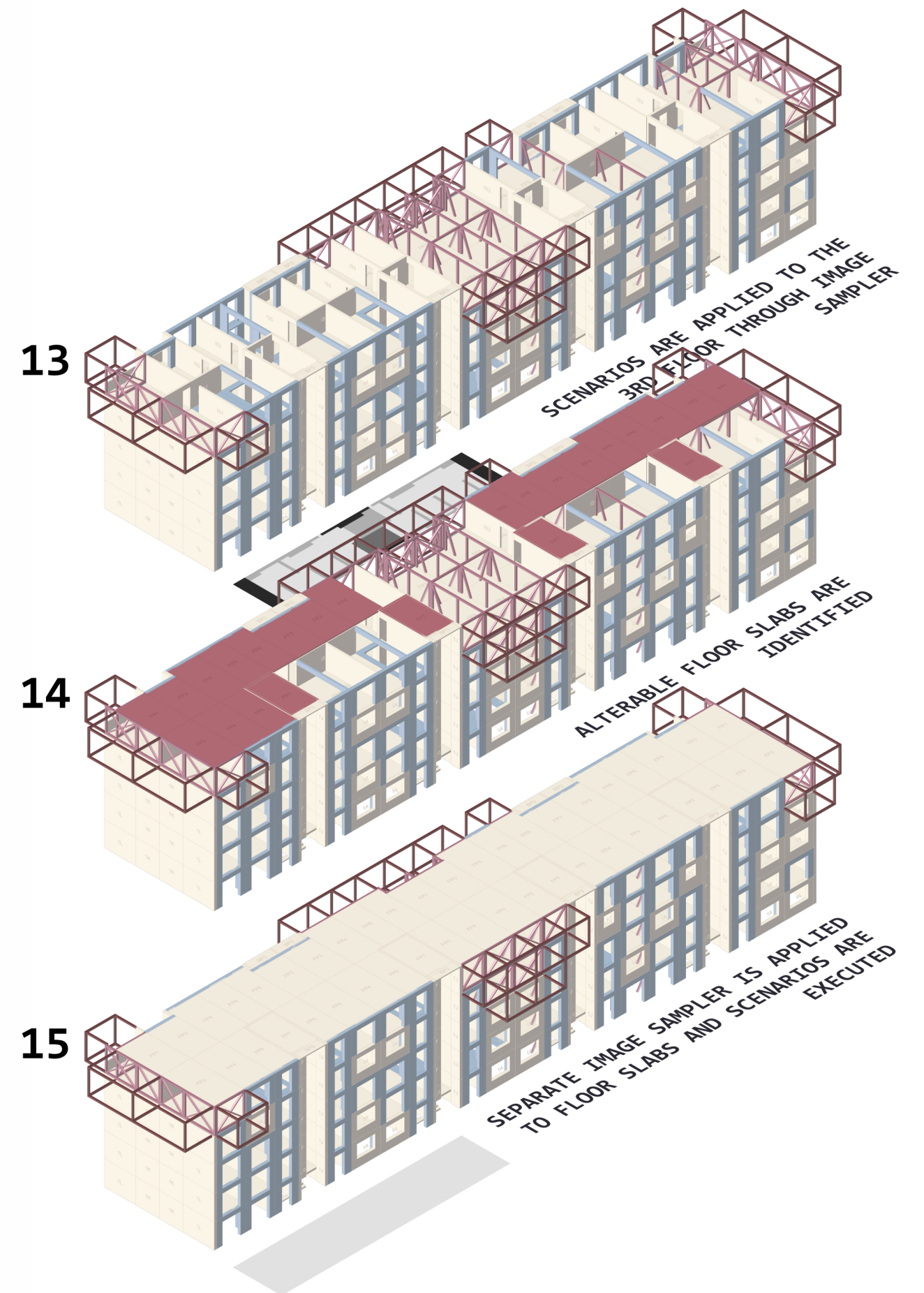
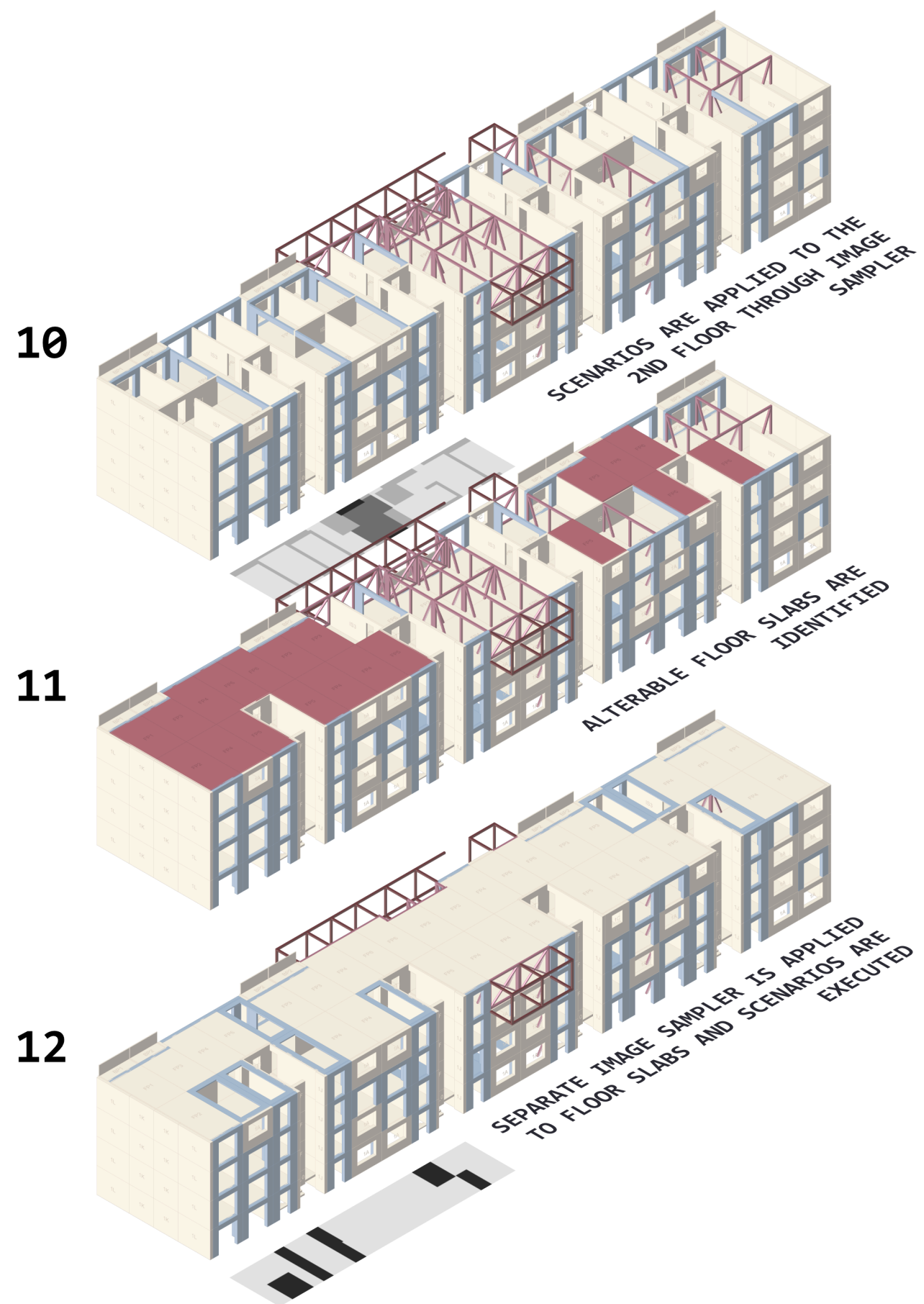
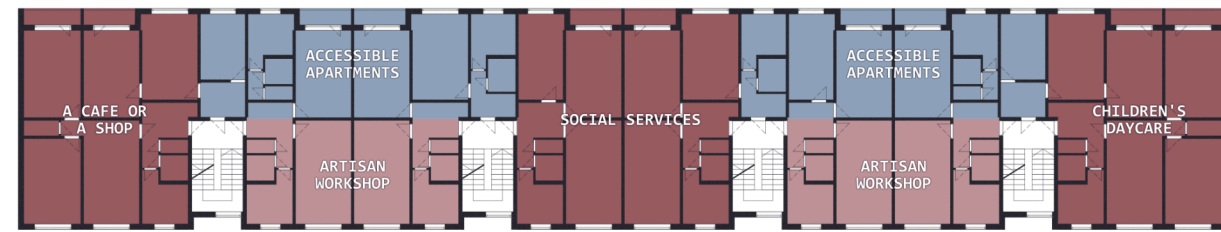


Figure 83. Algorithm application on the 4th and 5th floor elements for Experiment 5



GROUND FLOOR PLAN



2ND FLOOR PLAN



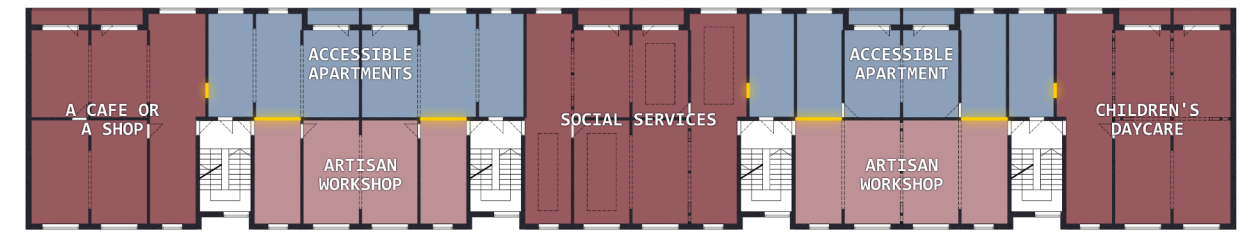
3RD FLOOR PLAN



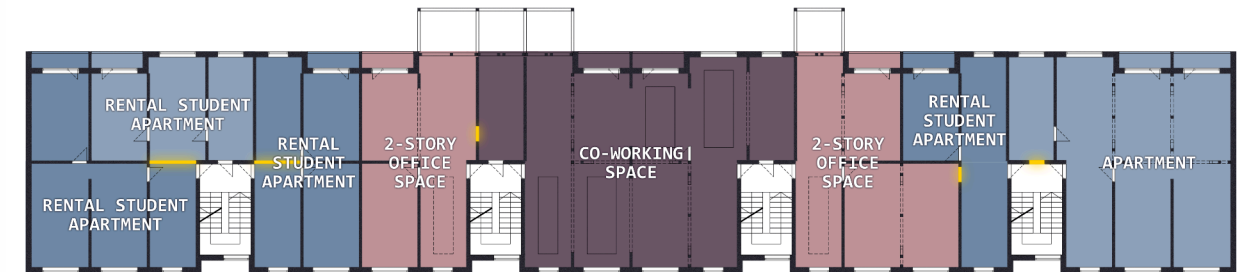
4TH FLOOR PLAN



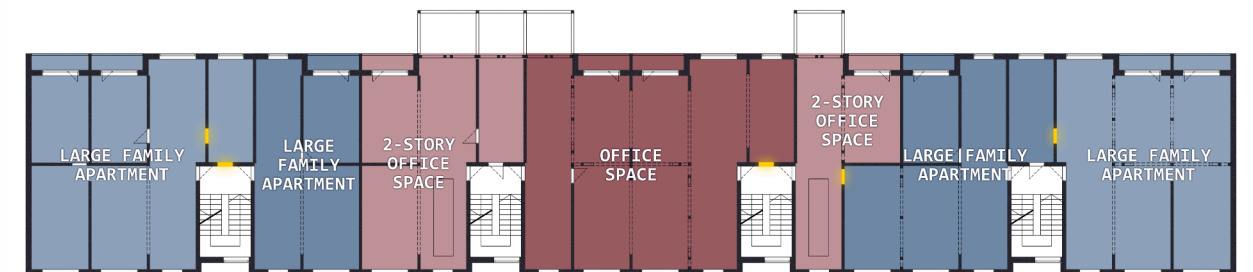
5TH FLOOR PLAN



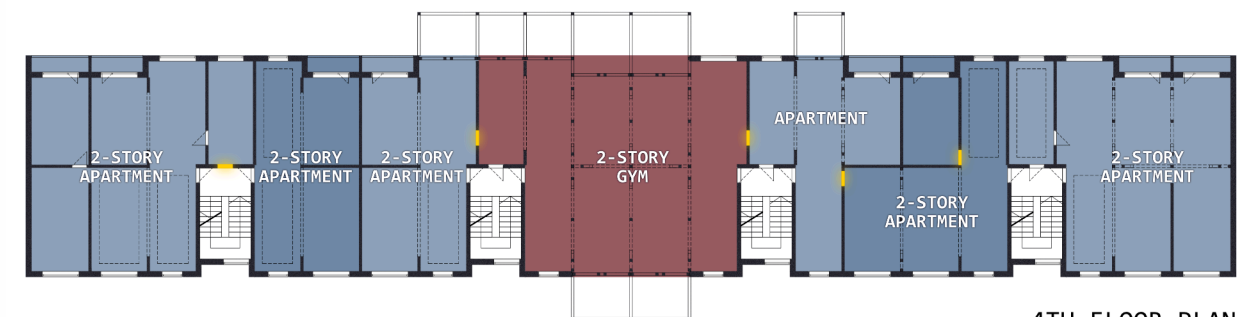
GROUND FLOOR PLAN



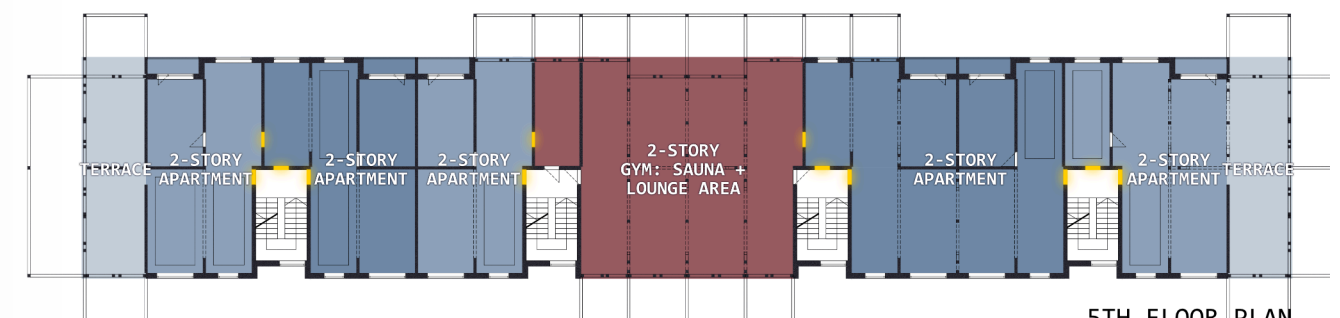
2ND FLOOR PLAN



3RD FLOOR PLAN



4TH FLOOR PLAN



5TH FLOOR PLAN

Figure 84. Original functional diagrams for all building floors (left) and adjusted functional diagrams after the floor plan extraction from the algorithmically altered building (right) for Experiment 5

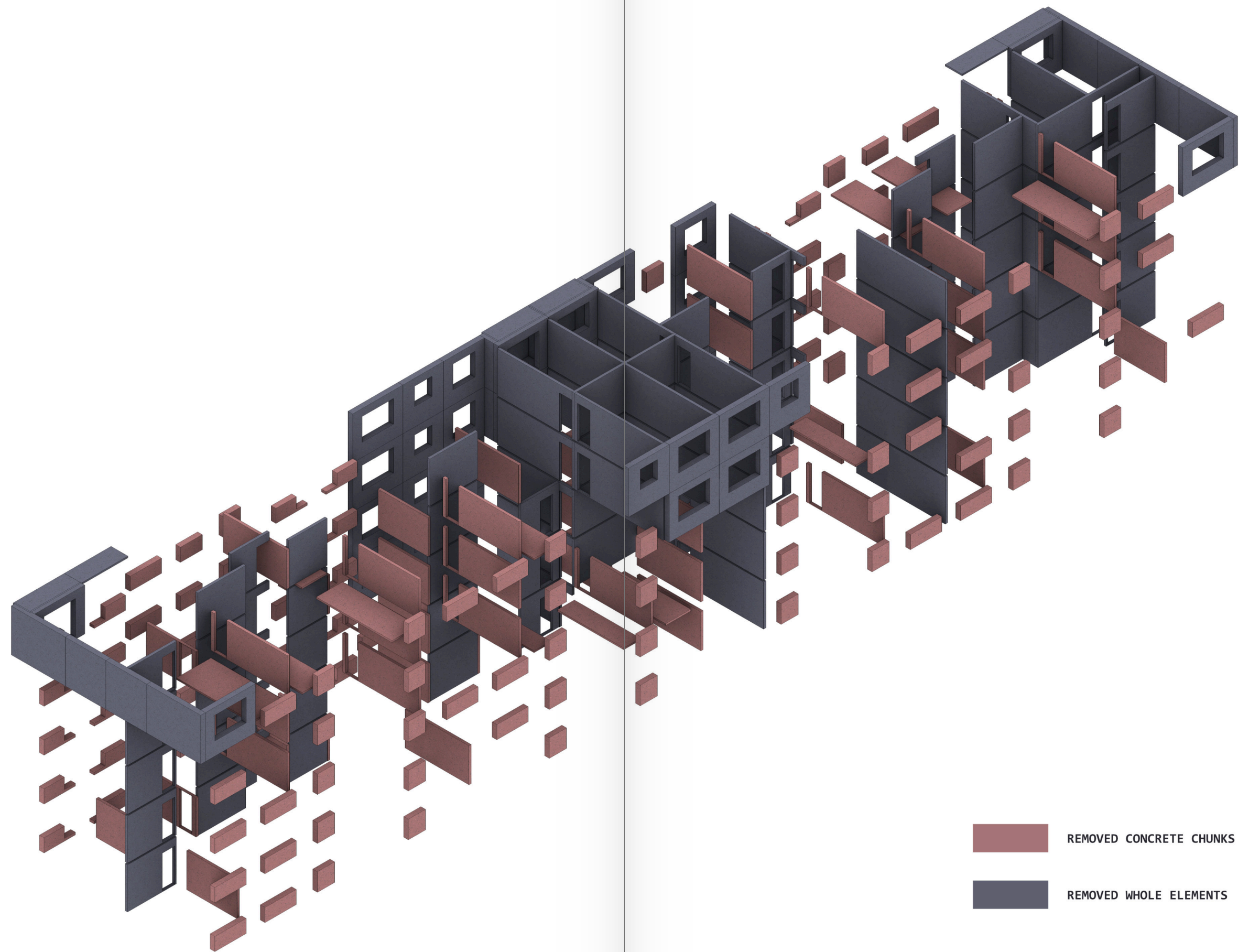


Figure 85. Isometric view of removed elements in Experiment 5, applicable for reuse in further architectural synthesis

8. ARCHITECTURAL SYNTHESIS

This section focuses on the architectural synthesis derived from the algorithmically generated output. The architectural synthesis within this project was carried out to provide proof that the created system works in a real-life application, and to enable reflections from an architectural standpoint. Therefore, the synthesis described in this section is therefore not to be viewed as a finalised architectural proposal, but rather as a proof of concept demonstrating that the system can work on a real building and can be used as a tool by an architect. This section also reflects on the architectural and urban impact of chosen iteration.

8.1 TRANSLATING THE ALGORITHMIC OUTPUT INTO AN ARCHITECTURAL OUTCOME

As established previously, the system functions as a tool for the architect and not vice versa – it is designed to assist in implementing changes according to specific vision, while handling the threat of progressive collapse by calculating the dependencies and applying the forced scenarios where necessary. While the previous section discusses what happens “before the system” (the initial user input), this section focuses on what happens “after the system” – how an architect can handle the outputs generated by the tool.

Fundamentally, the output of the system directly depends on the user input, therefore, the desired tweaks and changes should be first implemented during the grayscale image creation process. If the scenario application implemented by the Grasshopper script and grayscale image sampler does not fit the architectural intent, it can be adjusted by altering the grayscale image of the target floor. However, if an unwanted alteration happens because of a forced scenario application, lower floor grayscale images must also be reviewed in order to alter the target floor outcome.

There are also numerous user-defined values and choices within the script itself, discussed in detail in Section 6, such as the dimensions for new openings, dimensions of new frames, and decision-making nodes for the variable element scenarios. However, the final output geometry of the script should always be treated as a preliminary sketch of the architectural outcome, rather than a finalised building geometry. This is why the experiments in Section 7 within this thesis are depicted in greater detail than the architectural synthesis in Section 8: the project aims to showcase the different possibilities of the tool application rather than a finalised architectural vision. The synthesis was executed as proof that the system can the discussion and reflection of the “after the system” process involving the architect. The subsequent chapters therefore describe how the workflow from the algorithmic output to the final architectural synthesis was executed for this project.

The script-generated outcome of Experiment 5 was baked as static 3D geometry, separated by layers denoting modified, unmodified, and removed elements, alongside new frames and the new exoskeleton. This section elaborates on the detailing of the finalised building geometry to prove that the system created within this thesis can generate functioning architecture.

The first step of this phase was updating the floor plans. The functions and their boundaries created within Experiment 5 were readjusted according to the new geometries generated by the algorithm. Locations for required new elements were identified within the plan dimension to create placeholders for the potential reuse of the removed elements. The new outdoor spaces created by the addition of the exoskeleton were analysed against the new programmatic functions and several new locations for additional exoskeleton structures were identified to create more outdoor spaces on the ground and second floors.

The next step was the adjustment of the 3D model itself. First, the exoskeleton and the exterior frames had to be refined. In this iteration the exoskeleton and all frames in contact with the exterior were designated as steel. The frames were then adjusted to fit seamlessly with the remaining modified and unmodified elements and attach to the existing structure. The profile selected for this iteration was an I-beam. However, precise detailing on the attachment of this frame to the building – specifically regarding thermal barriers and structural connections – was omitted, as it falls outside the scope of this thesis and would be detailed in further phases. Furthermore, for the terraces created on floor 5, the exoskeleton was omitted, as that would unnecessarily alter the geometry. Instead, only the primary frames that replace the removed exterior walls were retained and detailed to ensure lateral and parallel bracing of the unremoved elements. Finally, a decision was made to extend the exoskeleton in the middle of the building one floor above the roof level to create a rooftop terrace. This could function as a communal outdoor space for residents, with access from the outside via an elevator or through the staircases, requiring the addition of two more flights of stairs.

Parallel to the exterior refinement, the interior frames were also adjusted. In this iteration, they were designed to consist of mass glulam timber, as it is more lightweight than steel and has warmer, softer aesthetic properties for interior spaces. Because these elements have no contact with the exterior environment, they can safely be made from timber. However, in terms of structural strength, precise load calculations should be carried out in future developmental stages to ensure these timber frames can actually support the imposed loads. The frames were modified and detailed according to their specific spatial situation – diagonal supports were only kept in places where they were supporting a remaining modified or unmodified wall; otherwise, they were removed to clear out more interior space.

Subsequently, the modified elements, specifically the floor slabs, were refined. As the modified floor slabs were created through the use of the script with an additional image sampler, they were already located correctly to ensure two-storey spaces, but the sizing of the openings had to be adjusted to accommodate vertical connections. Most of these openings were made smaller to ensure sufficient circulation space to move around the new staircases and atriums.

Next, all locations where new horizontal and vertical elements were needed were identified. These were modelled in a separate layer to then compare them against the library of removed elements and enable material reuse, aligning with the core thesis concept of *self-cannibalisation*.

Following this, the vertical connections themselves were modelled. As accessible apartments were created on the ground floor according to the functional diagrams, universal access had to be resolved. The existing staircases themselves are not accessible and are very difficult to retrofit, because the floor level of the ground floor is at +1.2m. Therefore, a decision was made to ensure accessibility to these apartments through the exterior, utilising the new exoskeleton and its integrated terraces created within. Ramps were created for accessibility from the outside. Additionally, the new café and daycare at the ends of the building were given separate staircases for their terraces to create the possibility for an autonomous entrance. Subsequently, the vertical connections within the interior spaces

were created as staircases. Finally, an elevator was added within the exoskeleton to ensure accessibility to most public spaces.

Finally, it was necessary to ensure that the building was fully enclosed, therefore all glazing and entrance elements were modelled. Vertical façade spaces where no new or reused elements were placed were filled with full-height glazing within the frame. Other windows were modelled according to the opening sizes. Similarly, interior doors and entrances were modelled according to the opening sizes as well. To conclude the architectural synthesis, the railings for the new terraces were modelled according to the boundaries set in the plans.

Upon finalisation of the building geometry, it was integrated into its urban context. The original site of the reference building within Iļģuciems was retained. The plot is bounded by a major street along its eastern edge and a minor access road wrapping around the north and west. The original orientation of the building was preserved, maintaining the primary entrance nodes of the northern façade while capitalising on the southern exposure for the main terrace blocks. Pedestrian paths around the building were redesigned to align with the new architectural organisation. A communal area, featuring barbecue facilities and outdoor seating, was situated at the western end of the site. To support the ground-floor daycare, a dedicated playground was introduced nearby, complemented by a larger recreational area and an outdoor gym situated across the main path. Furthermore, community garden plots were integrated along the southern façade of the building. Because the original building occupies a generously spaced land plot, the site offers substantial opportunities to cultivate active, community-oriented landscapes that support the building's new programmatic functions.

8.2 TECTONICS AND MATERIALITY

As mentioned, for the structure of the exoskeleton and the interior grid, a lightweight structure was chosen. Because structural insertion is the highest level of intervention, as it deals with full element removal and replacement, and the intention of these scenarios was to open up the spaces on the interior (Scenario C) and towards the exterior and exoskeleton (Scenario D), a logical decision was to make these structures lightweight to ensure that the spaces actually become open. As discussed, the script itself only outputs preliminary geometry; therefore the decision regarding the actual materials is made in the architectural synthesis phase. However, as the building is a decaying structure, and one of the main current concerns its structural integrity, the decision to insert a lightweight frame structure instead of, for example, concrete, was logical from both the architectural and the structural standpoint.

To address the visible propagation of the scenarios C and D – the element replacement – the decision to create this system with a forced scenario application stemmed from a combination of factors. Firstly, the logic of physical execution: if a wall panel is removed, all the panels above it that this target wall supports, have to be removed as well. Second, the choice of the lightweight frame insertion itself: even though not all the walls are solid reinforced concrete panels, they are still quite massive and dense, therefore, utilising a lightweight frame to support these structures was not structurally viable. Third, in addition to the structural logic of having heavyweight elements at the bottom and lightweight elements atop of them, this is also a semantic and architectural decision, enabling the preservation of the lower part of the building as a rigid concrete structure while adding the more lightweight elements towards the top floors, creating a visually balanced assembly. Therefore, the decision of vertical forced scenarios and the decision of lightweight frame insertion go hand in hand with each other, and represent a cohesive decision from both a structural and architectural standpoint.

8.3 STRUCTURAL, ARCHITECTURAL AND URBAN CONSEQUENCES

As this project sits at the intersection of structural alteration, architectural reorganisation, computational design, the mitigation of urban deficiencies, material reuse possibilities and addressing local building expiration, this chapter analyses the thesis from these standpoints.

Structural

The structural reorganisation proposed within this project is inherently experimental, and further real-world tests would have to be executed to be deemed realistic. During the project, specific structural loads were not calculated, however, the main aim to preserve the integrity of the structure was met on a computational level. The main structural concern of progressive collapse was addressed within the creation of the dependency script and the new added structures follow the rigid, original grid of the building. However, the main question that remains open is whether any of these alterations are realistic within the existing building at all as evaluating that would require empirical tests and analysis of the structure itself.

Architectural

The architectural and spatial reorganisation executed within this project can be evaluated as successful. The different scenario applications allow for the creation of open spaces and injections of new functions that were impossible in the original building. However, there are several downsides to this architectural approach as well – as the building is transformed from a rigidly mono-functional to a highly multi-functional building, it raises questions of privacy for the residents, and consequently, safety. These concerns could be mitigated in future iteration by strategically separating circulation paths for public and private functions using the exoskeleton, or by enforcing strict functional zoning rules within the image sampler input creation phase. Moreover, the architectural outcome highly depends on the architectural synthesis phase and decisions made by the user. In the case of the synthesis done in this project, the final weatherproofing enclosing layer of the building was not taken into account, as it exceeds the boundaries of the project, however, the decisions made for this

could affect the outcome of the outer form and aesthetic of the building depending on the finishes chosen. In terms of interior spaces, the goal was met, and it was proven that spaces very different from the original building can be created through the use of this tool.

Computational

For the computational framework, a relatively simple workflow was chosen, but as it had to address structural dependencies and a large number of objects at once, it became complex rapidly. The dependency script utilises Anemone loops within Grasshopper environment, therefore it is quite computationally heavy. Additionally, the procedure of inputting the image to the image sampler and the application of the scenarios, as well as the dependency calculation, has to be done five times – for the five floors of the building, therefore the script becomes heavy and processes slowly. Moreover, as it is a Grasshopper script, it is not very user-friendly, therefore some adjustments could be implemented. Finally, as it is a tool created by adding and modifying parameters “as you go” rather than built linearly from start to finish, there are definitely some redundancies that could be optimised.

Urban

As identified in Section 2, the biggest urban deficiency within Iļģuciems is the lack of public and community spaces, therefore, the functional injections for Experiment 5 and the final architectural synthesis were created to implement these functions within the building. The implementation of the new public functions within the previously mono-functional building in combination with the added exoskeleton and removed walls opens up the building towards the city, blending it better with the neighbourhood. The public functions were specifically designed to be more open towards the outside, and with the added expansions the building sits harmoniously in the urban context. However, the raised ground floor prevents the building from being fully integrated within the neighbourhood from both an accessibility and a streetscape perspective. Nevertheless, the goal of adding more community and gathering spaces within the neighbourhood was achieved and the building functions as far more open environment to the outside world than beforehand. Additionally, the added community spaces on the exterior help blend

this building within the existing neighbourhood while still adding additional value to the urban environment.

Reuse

The reuse goals were partially met. While the intention of reusing the removed elements from the building within the same building, enforcing the “self-cannibalisation” approach is met and somewhat integrated within the project, further detailing on exactly which removed elements are placed where within the building should be done to solidify the intent. Although the computational script successfully extracts the inventory catalogue and the needed new elements and their dimensions are identified within the synthesis phase, resolving the actual structural joint detailing for these “self-cannibalised” components remains a complex tectonic hurdle that would require subsequent engineering phases. Moreover, no calculations have been made to specify how much material is removed from the building and reused, but that could easily be the next step as there is a catalogue of removed elements encoded within the tool and the needed new or reused elements are also specified.

Local building expiration

This project and architectural synthesis successfully prove that Soviet prefabricated concrete panels can be adapted to modern life and functions outside the residential realm. This project serves as proof that there is an alternative to demolition and conventional renovation, however, to bring this project to a real-life scale, further detailing in all the sections mentioned above should be executed.

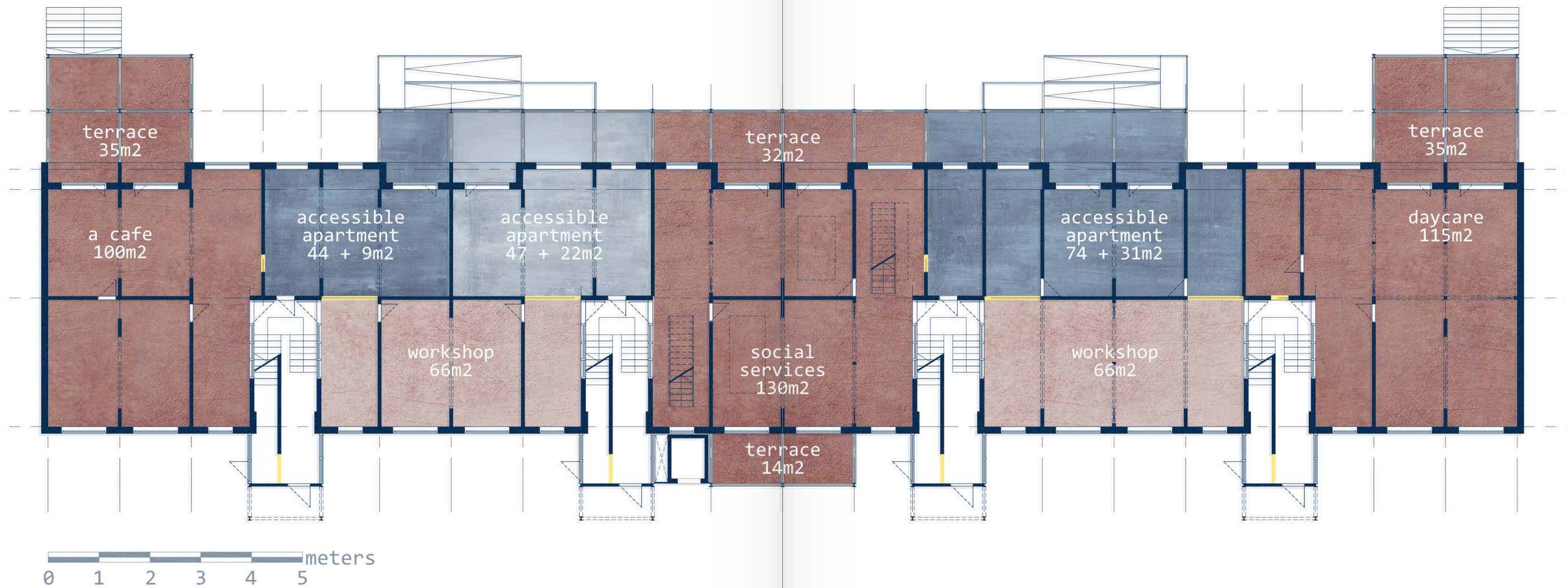


Figure 86. Building ground floor plan after architectural alterations and adjustments

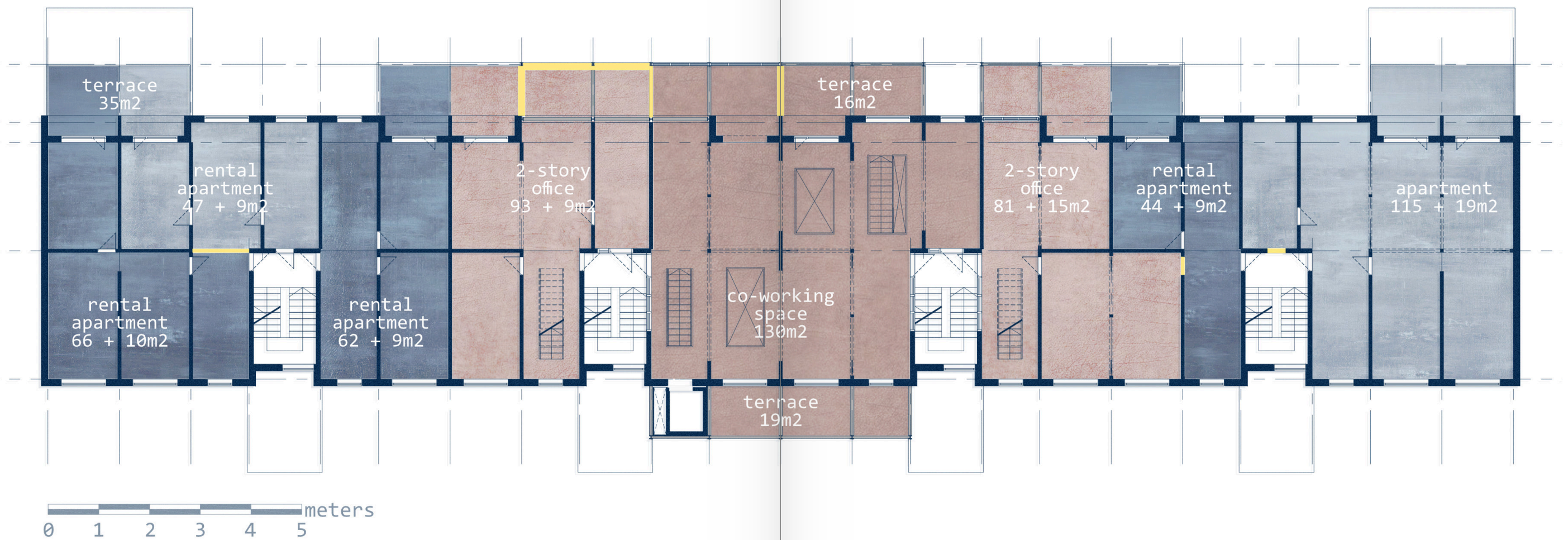


Figure 87. Building 2nd floor plan after architectural alterations and adjustments

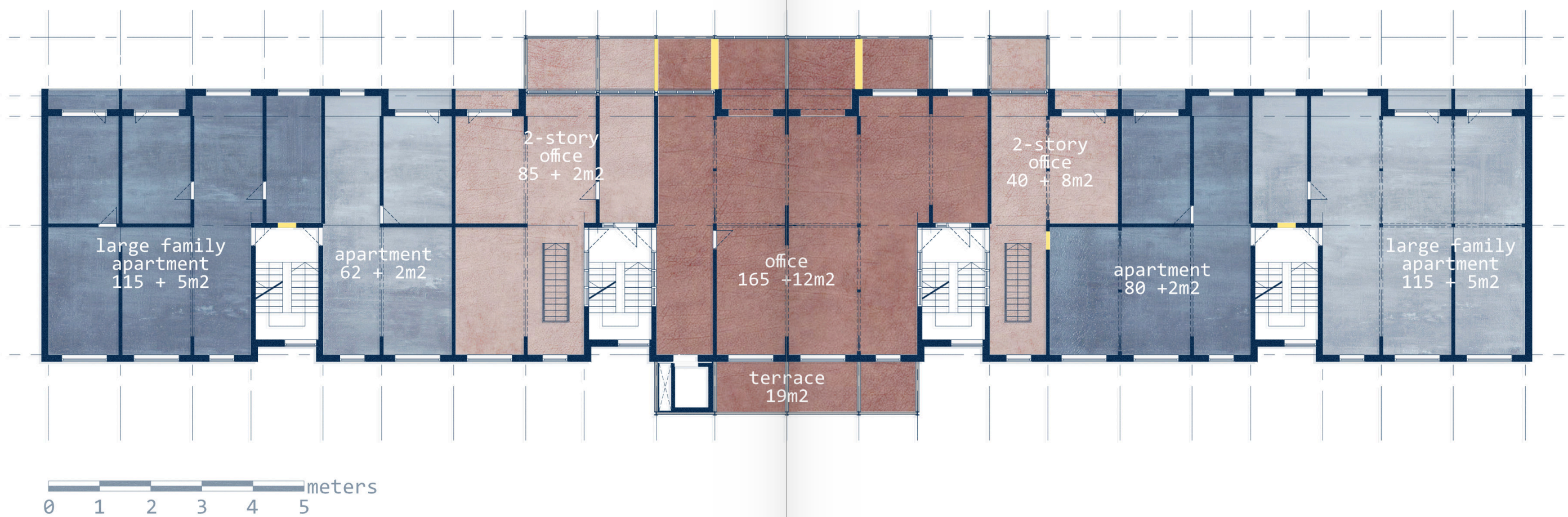


Figure 88. Building 3rd floor plan after architectural alterations and adjustments

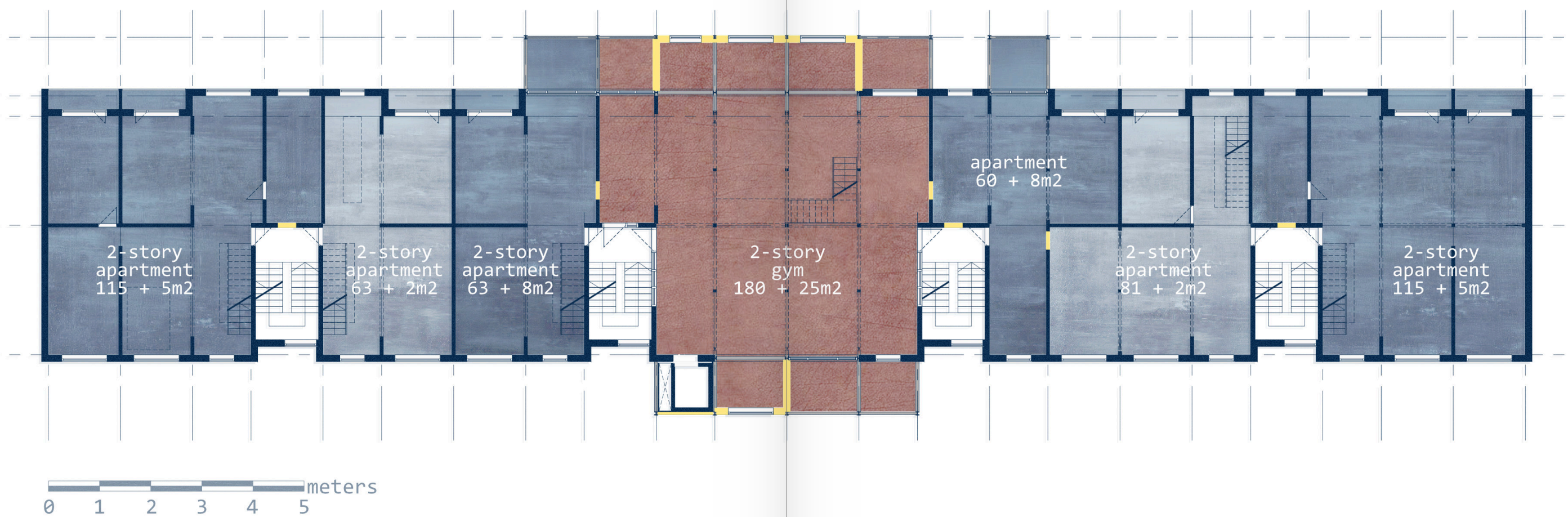


Figure 89. Building 4th floor plan after architectural alterations and adjustments

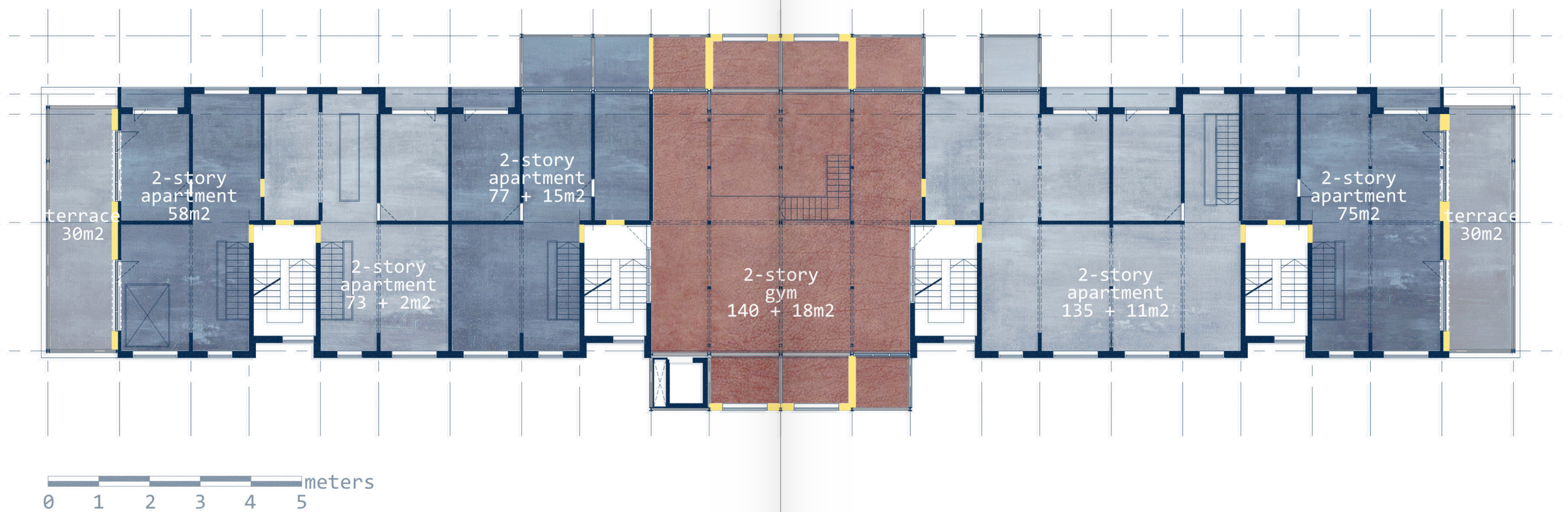


Figure 90. Building 5th floor plan after architectural alterations and adjustments

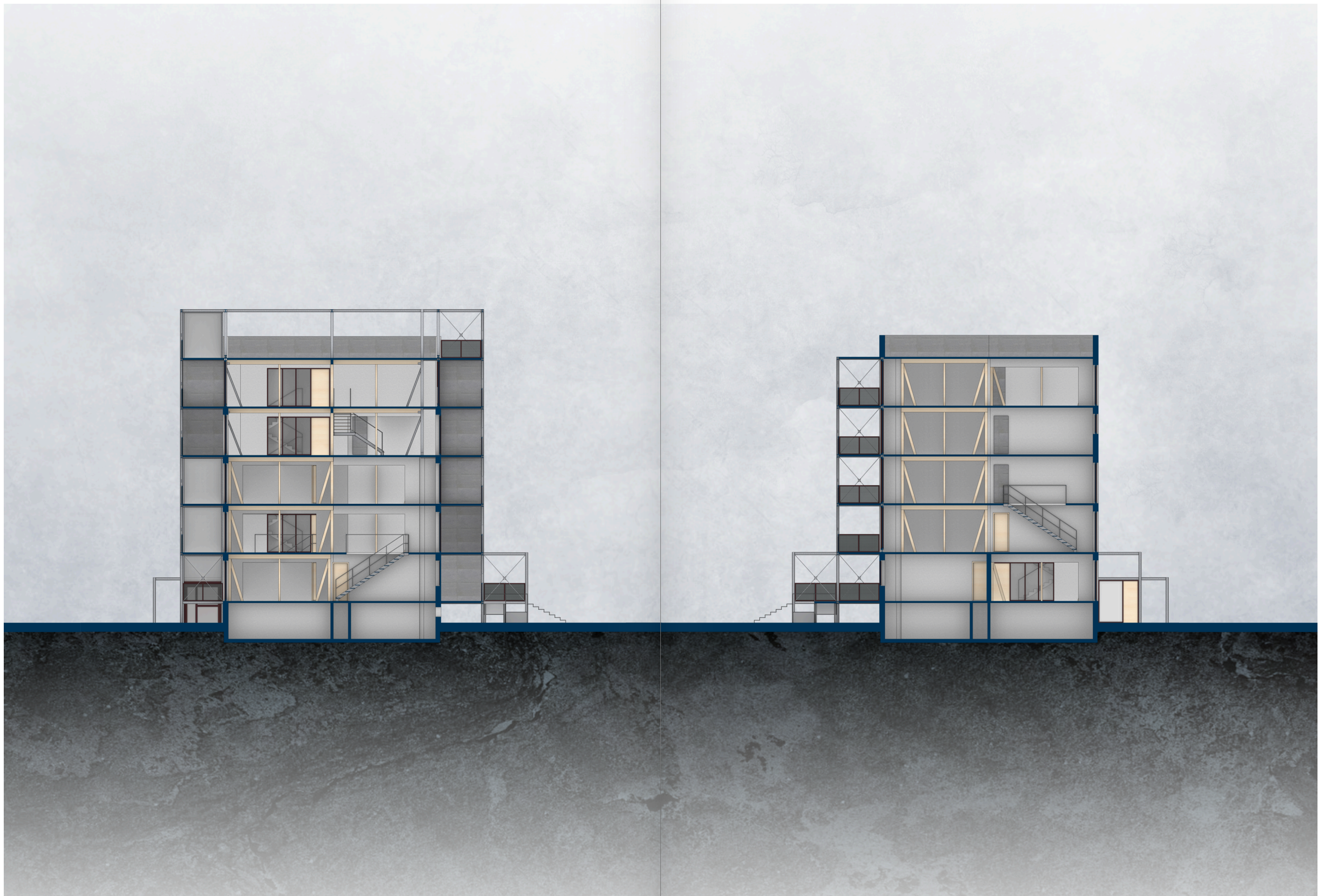


Figure 91. Building cross-sections after architectural alterations and adjustments



Figure 92. Building longitudinal section after architectural alterations and adjustments



Figure 93. Building West and North elevations after architectural alterations and adjustments



Figure 94. Building East and South elevations after architectural alterations and adjustments



Figure 95. An interior view of a two-storey apartment

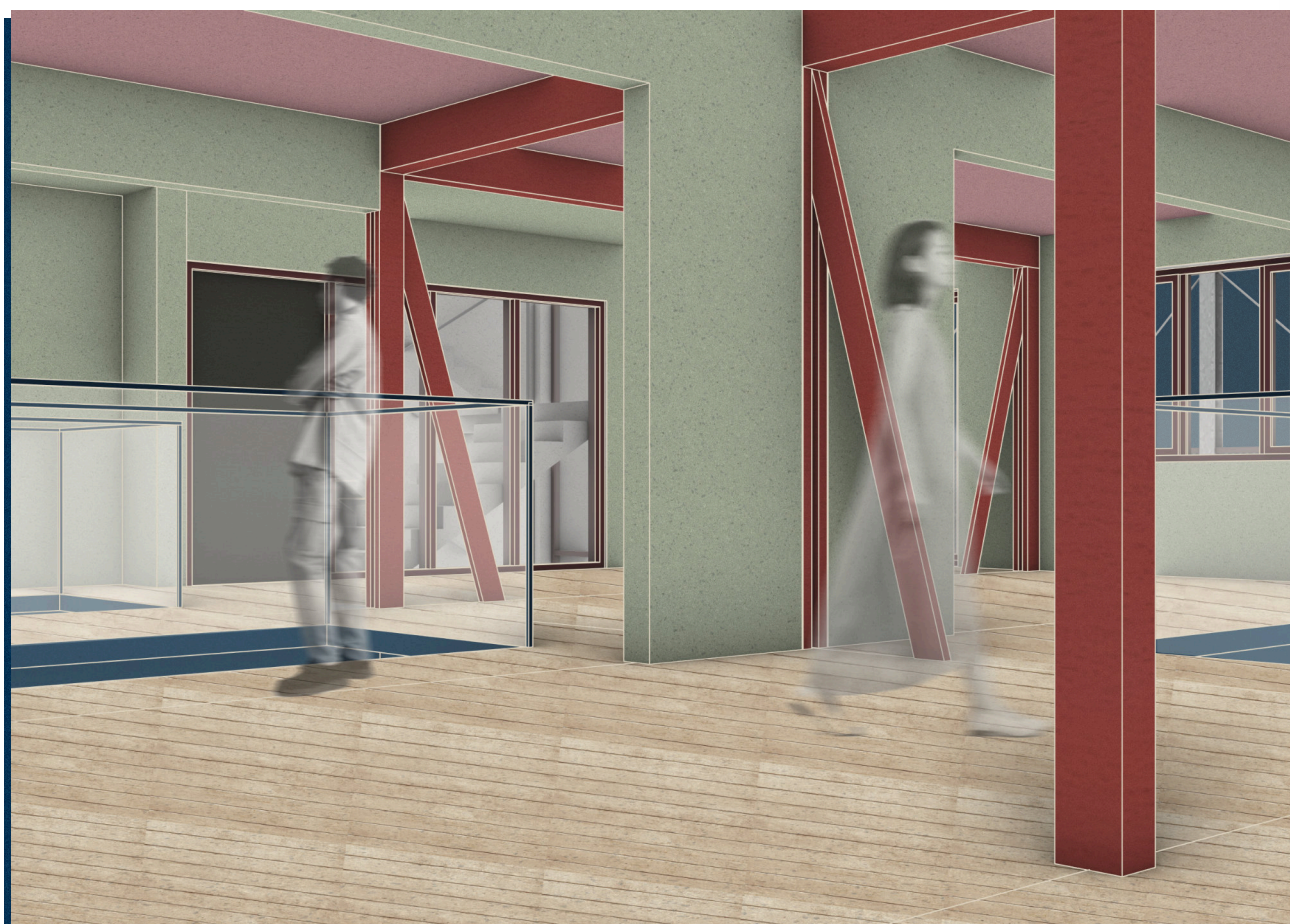


Figure 96. An interior view of the co-working space



Figure 97. An interior view of the two-storey gym

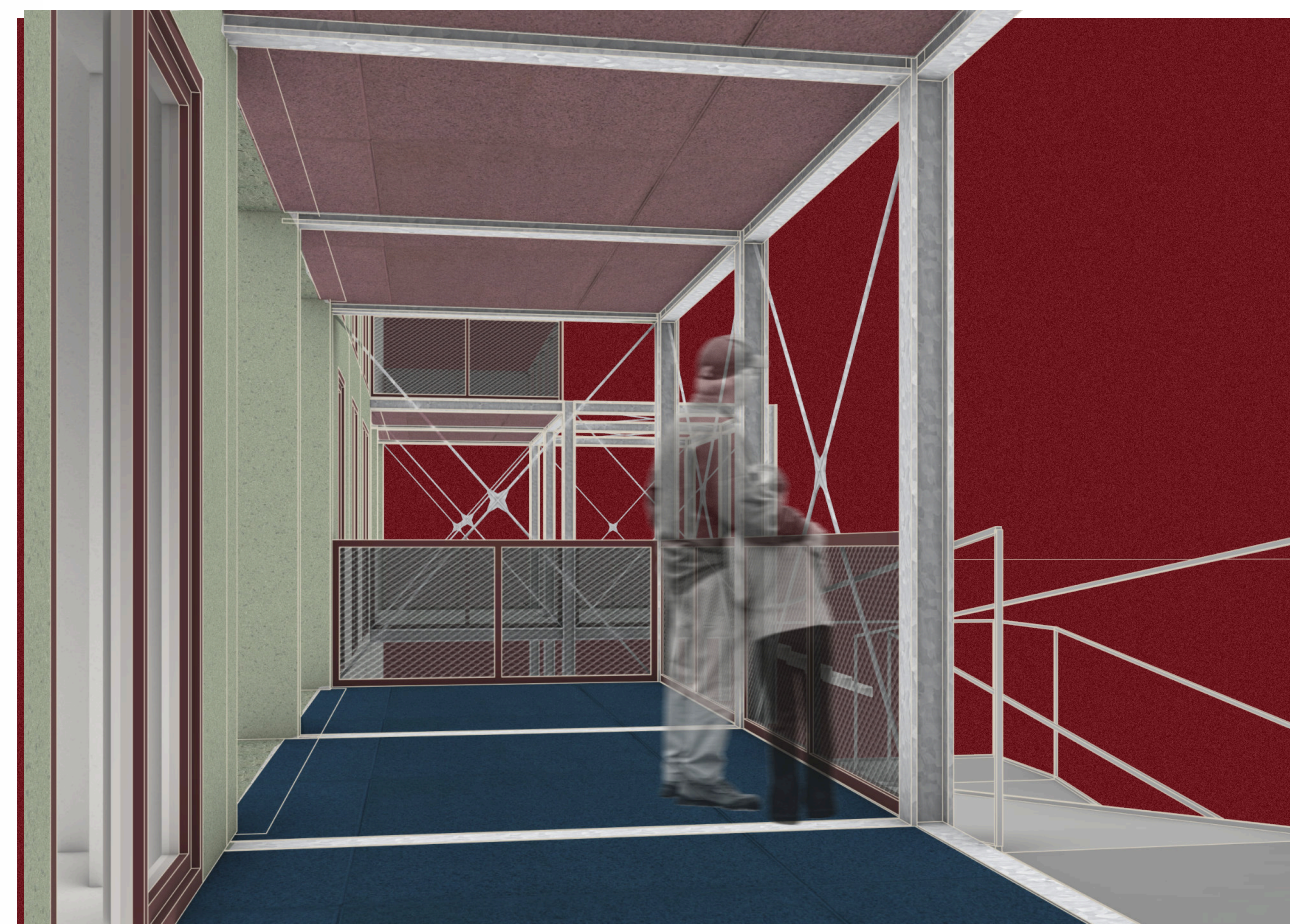


Figure 98. An exterior view of the exoskeleton-based terrace

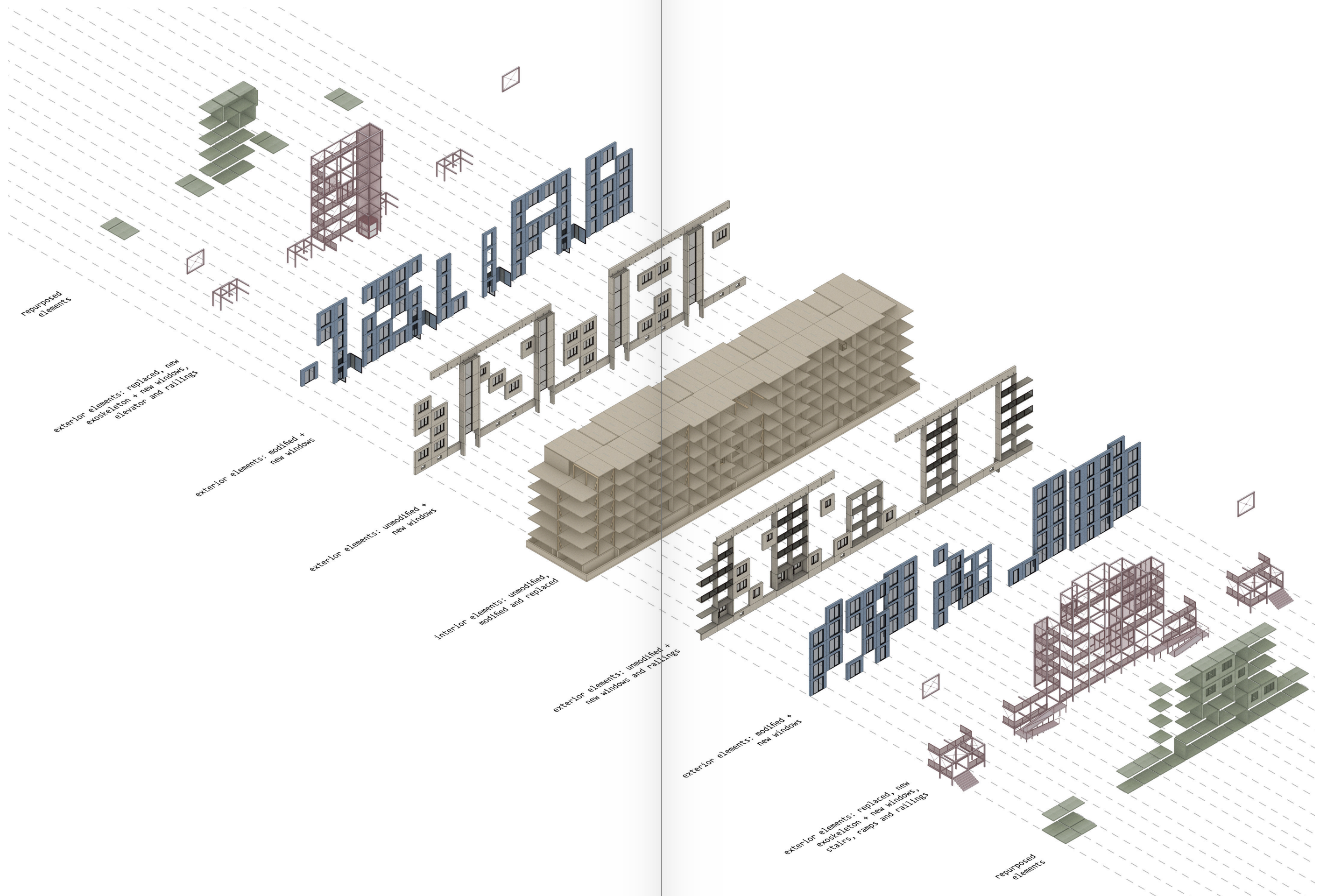


Figure 99. An isometric diagram of distinct unmodified, modified, added and repurposed elements

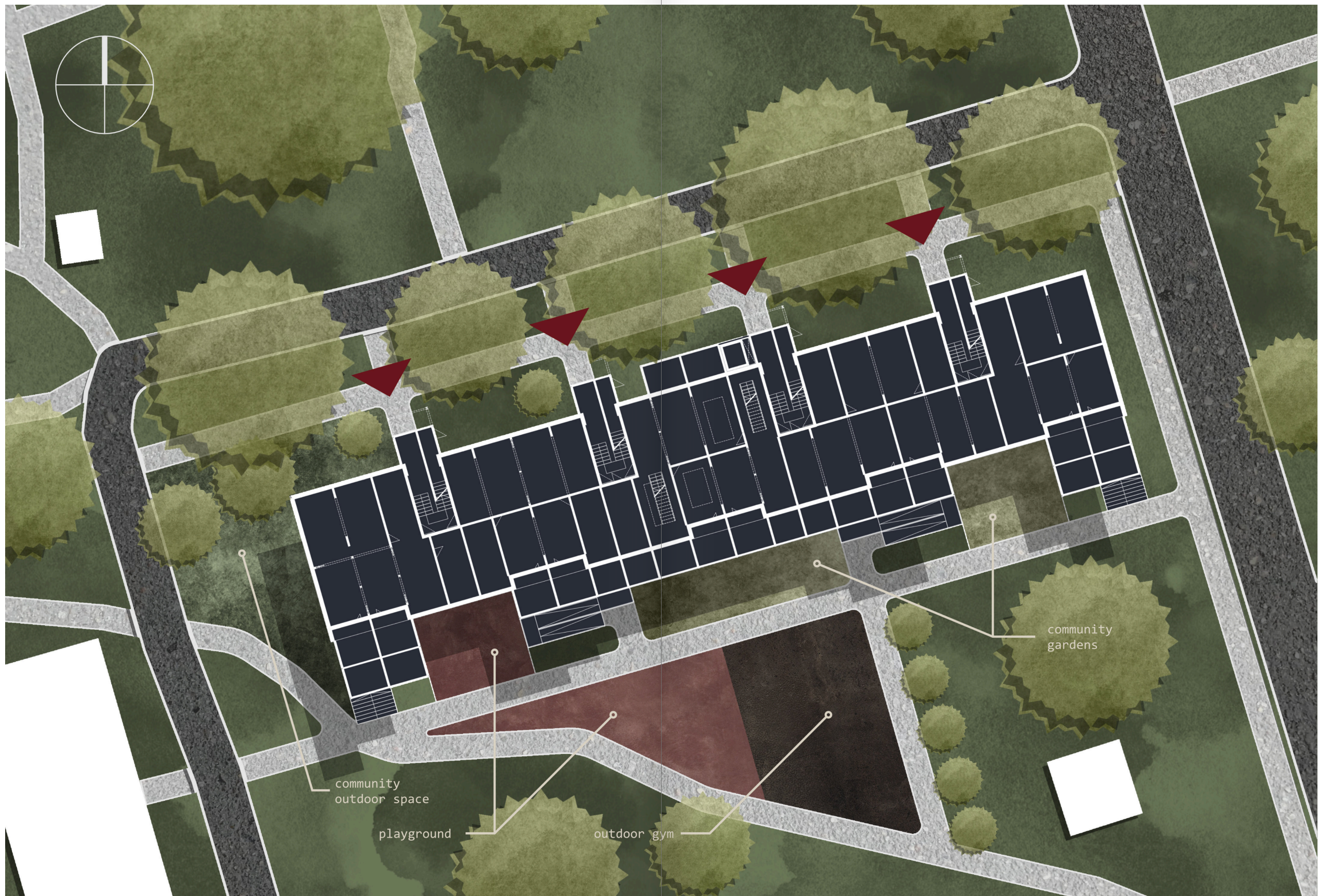


Figure 100. Site plan illustrating the building within the reorganised community landscape



Figure 101. Images of the physical model showcasing a section of the transformed building



Figure 102. Images of the physical model showcasing a section of the transformed building - exterior

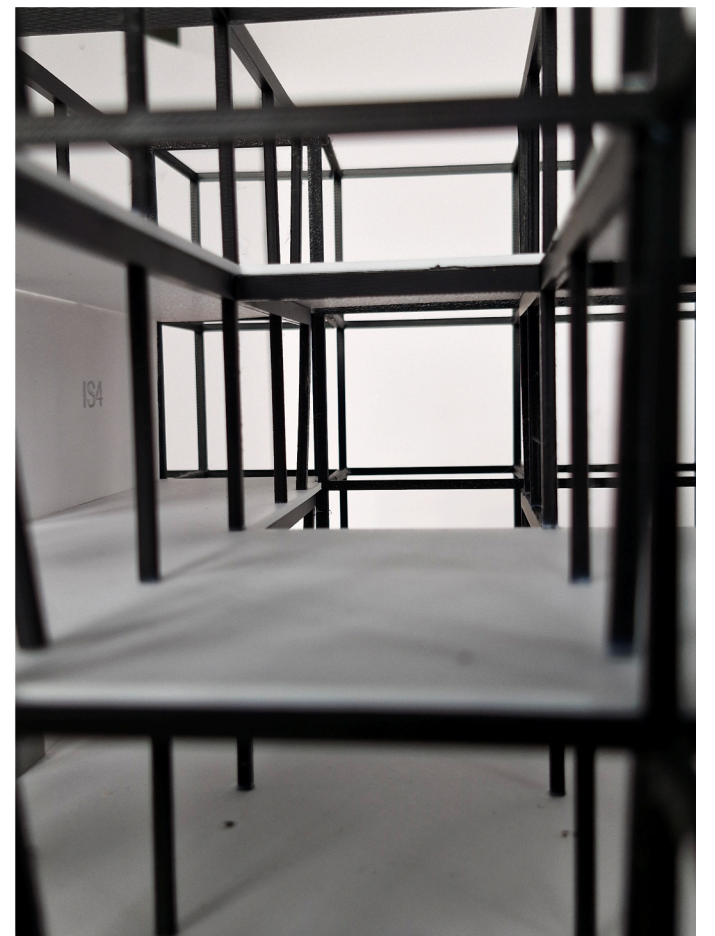


Figure 103. Images of the physical model showcasing a section of the transformed building - interior

9. CONCLUSIONS AND REFLECTIONS

9.1 CONCLUSIONS

Resolving the dual crisis

This thesis set out to address two intersecting crises: the global ecological impact of the construction industry through the exclusive use of raw materials and entirely new components, and the local structural and technical expiration of Soviet mass housing in Riga. While the project's workflow leaned more towards the local issue, as a specific neighbourhood was inspected and a heavy focus on one particular local building typology was maintained. However, the global emergency was addressed within this local scope as well, developing a system of building "self-cannibalisation" and structural reorganisation within the existing building boundaries. Therefore, this thesis demonstrates how buildings can function as material banks, even if the building itself is both the donor and the receiver.

Furthermore, this research successfully demonstrates that demolition, the "tabula rasa" approach and conventional renovation are not the only ways to deal with deteriorating buildings that have become inconvenient. In the case of Soviet mass housing, several spatial issues were identified during this research as well, such as cramped "pixelated" layouts with small cells and rooms. This project therefore aims to break the clear cell pattern to enable adaptive reuse and the introduction of new functions within these originally strictly mono-functional residential buildings. Additionally, it demonstrates how the building can be expanded outwards by adding a steel exoskeleton and retrofitting the removed elements within this structure to create new exterior boundaries.

The success of computational methodology

The computational tool for the structural and spatial reorganisation of the 1-464A series developed within this thesis proved to be successful in bridging the gap between structural possibilities and architectural necessities. The tool was heavily focused on identifying the structural dependencies between elements to prevent a progressive collapse of the building if an element of the

system is removed. This enabled the creation of a system that allows the user to feed an architectural intent and a functional programme into this tool for each building floor, while simultaneously calculating the risks of progressive collapse and dictating the floors above based on the floor below.

The experiments conducted with the tool successfully prove that it does not function autonomously as an independent geometry generator, but rather as a tool that can be used by an architect to explore different spatial possibilities while the algorithm calculates the structural risks. Essentially, the grayscale sampling acts as a digitised sketching mechanism requiring architectural interpretation, rather than an automated pipeline. Therefore, the tool serves the architect and not vice versa, which was the exact intention behind its creation.

Architectural and structural synthesis

The final experiment that resulted in the architectural synthesis proved that deep spatial reorganisation is possible within the rigid parameters of the Soviet prefabricated concrete housing typology. By carving out double-height voids, replacing walls with structural frames and creating larger openings within the wall panels, therefore merging claustrophobic units and rooms, the building's spatial quality was elevated to meet modern standards. Crucially, the unusable concrete chunks harvested during this extraction are relegated to site landscaping, while intact panels are accounted for and reserved for the building's own reorganisation. This same system also allowed for the introduction of new functionality within this previously strictly residential building block. The architectural synthesis proves that the system works and that a functioning building can be produced through its use.

Social and urban impact

Within the urban analysis conducted at the beginning of this research, it was identified that the focus neighbourhood – Iļģuciems – critically lacks community and cultural spaces. Although historically the idea of communal spaces has been frowned upon as it was strictly imposed on the inhabitants during the Soviet period, the times, and the generations bearing that trauma, are changing and it has become modern again,

especially when looking at Scandinavia as an example. Communal spaces always raise questions of "if everyone is responsible for something equally then nobody is responsible at all" or the tragedy of the commons, but resolving this social aspect is not exactly within the scope of an architectural thesis. Nevertheless, integrating a co-working or co-gardening space within a residential unit addresses a highly relevant contemporary desire for shared amenities.

Therefore, the building, which was originally completely rigid, uniform and enclosed from the outside world, is opened up; public spaces are introduced on the ground floor and in a central section of the building to connect not just the apartments, but the whole building and its inhabitants to the outside world. By transitioning from isolated cells to integrated co-working spaces, cafes, open-layout apartments, and expanded sightlines, the architecture effectively turns a dormant sleeper-district into an active community hub.

Future applicability and outlook

While this thesis primarily focused on one specific building within the Soviet residential housing series 1-464A in Iļģuciems, the computational logic and philosophy could be scaled up to different prefabricated concrete panel buildings as well. Of course, a structural diagnostic of the building would need to be carried out again to determine the element relationships, but the ideology is highly scalable. Therefore, this thesis proves that demolition is not the only way, nor is conventional renovation that does not necessarily improve the living conditions within these buildings. As Soviet residential building deterioration and expiration is a persisting issue in all post-Soviet countries of Eastern Europe, this methodology could easily be applied with a different local focus.

Moreover, this project operates on the prefabricated element scale, and does not go into detail regarding thermal insulation, connection details and the physical execution of the proposed alterations. That is definitely a topic that should be explored in more detail in the future, as it would enable bringing this proposal into the real-world dimension.

9.2 DISCUSSION AND REFLECTIONS

A persistent issue in Latvia, particularly in Riga, is the stagnation in renovating the Soviet residential building typology, even though the state as well as the EU offer partial subsidising of the costs. The reasons behind that are varied, but a primary factor I can identify from personal experience is the ageing population (a large percent of the residents within these buildings are elderly), which makes their funds very limited. Additionally, they simply do not see any reason to invest in something that will not change their living conditions that much. Because conventional renovation typically involves merely adding thermal insulation, replacements of windows, plumbing and heating systems, it may reduce the energy consumption and therefore bills, but does not alter the geometry and layouts of the apartments themselves. Therefore, while renovation is better than doing nothing at all, residents most of the time decide to opt out of that as well and the Soviet residential buildings keep decaying up until a point in the near future where they simply will become “beyond repair” and therefore uninhabitable.

To address this gap, the tool created for this thesis was developed to facilitate the spatial and structural reorganisation of Soviet precast concrete residential buildings through a scalable, series-wide application. It serves as a radical alternative to conventional renovation. Crucially, while surface-level renovation can be carried out while current residents maintain occupancy and ownership over their apartments, this fragmented model has paralysed systemic progress. Therefore, the framework created within this thesis approaches the problem with a different mindset: the proposed alterations cannot take place within current ownership and apartment boundaries. For this systemic adaptation to happen, all properties within a building would have to be consolidated by a single entity, perhaps a property development company, to treat the building as a cohesive structural system rather than a collection of isolated apartment cells.

Regarding the application of the tool itself, the framework was based specifically on the 1-464A series due to its prevalence in the urban environment of Riga. However, in reality, no two buildings are identical: as they were built, each assembly was adjusted to its localised context. Some feature attic spaces,

others do not; in some cases, there are windows at the “end” walls of the building, in others there are not. Furthermore, overall spatial arrangements of the buildings within the *microrayons* vary significantly – ranging from dense clusters to free-standing iterations. Therefore, even though these buildings were designed to be completely uniform across the city and the whole Soviet Union, in reality, there are still slight differences from instance to instance. Additionally, as discussed in Section 4.3, the series itself is defined by “end” and “middle” modules, which can be combined, expanded and arranged in countless configurations.

These different variations of the same system are exactly what make the framework created within this project universally applicable and scalable within this specific series. While for the case study within this thesis one specific assembly was selected to define the inter-element relationships, dependencies and alterations, the prefabricated element factor allows for the system to be expanded. No matter what the building’s final, existing configuration is, the key prefabricated elements and their relationships remain the same. As proven by the experiments described in Section 7, the possible outcomes are highly variable based on the user’s – the architect’s – input and vision. The outcomes are currently limited to the predefined element scenarios, and they are also limited by the orthogonal grid dictated by the existing building structural scheme, but even within this frame, there is still a great variation on what can be done with the building.

Ultimately, the created tool is definitely applicable as a system for the reorganisation of different buildings within the 1-464A series, rather than a tool for a singular project. While this system of spatial and structural reorganisation should not be applied to each and every building of this typology within Riga, the experiments prove that vastly different architectural outcomes are possible depending on the user input. One could argue that the tool implies a very monotone architectural style, but these buildings were originally designed to be completely uniform, and this system opens up way more possibilities for these rigid, strictly mono-functional residential buildings both in the architectural form and the spatial organisation than conventional renovation approach. The system still follows a rigid orthogonal grid

dictated by the original building, however the alterations proposed within this grid serve as an architectural bridge between the existing and the new. Instead of viewing the rigidity and strict order as a disadvantage, this system offers a way to expand the grid and work with what is already there. Furthermore, the decision to insert a lightweight timber frame structure instead of concrete was a structural necessity rather than just an aesthetic contrast: the existing concrete lower levels lack the load-bearing capacity to safely support additional heavy loads, making the timber endoskeleton and steel exoskeleton a forced engineering parameter. Essentially, the system allows for the expansion and alteration of the rigid, monotonous structural and spatial system of prefabricated concrete panel residential buildings by subtraction, replacement, reuse, addition, and reorganisation without disrupting the original system.

For a final reflection, and future outlook, there are several topics that were left out during this project. For example, this building typology famously has a very low-ceiling height, and the alteration of that could not be incorporated as the project worked with an existing building system. Additionally, the actual details and connections of the new frames were not solved within this project scope, and the same goes for the reattachment of the reused elements. Moreover, while the repurposed elements are currently placed in the building manually, future iterations could implement an expanded Grasshopper and Python script to computationally compare the needed elements to the removed element library. The plans are currently left in the functional zoning stage, whereas the original plumbing nodes should be preserved, and more detailing should be done to ensure that it is possible to do. The building should also be properly enclosed in terms of thermal, sound and weather – proofing, as the tool currently only operates on the prefabricated element scale.

10. ACKNOWLEDGEMENTS

I would like to acknowledge my supervisor, Gediminas Kirdeikis, and my examiner, David Andr  en, for their roles in the conclusion of this project.

Declaration of AI Usage

Text Editing: AI tools (specifically Google Gemini) were employed to assist with formatting, language refinement, and editing of this report. However, AI was not used for the generation of original text and ideas - all foundational concepts, arguments, and architectural conclusions are the sole original work of the author.

Computational Development: Generative AI was utilised for syntax troubleshooting and coding assistance within the Grasshopper and Python environments. The underlying structural logic, dependency definitions, and systemic architectural frameworks were conceived and directed entirely by the author.

11. TABLE OF FIGURES

Please note that all visual materials - including but not limited to photographs, diagrams, floor plans, and digital renderings - presented in this report are the sole property and original creation of the author. Any graphics or figures sourced from external materials are explicitly credited within their respective captions.

Figure 1. Comparison between the original building facade and a facade altered using the developed computational framework	4	Figure 10. Different series of Soviet period precast concrete panel residential buildings in I��guciems	17
Figure 2. Proposed six-phase digital workflow outlining the thesis methodology	6	Figure 11. Commercial building typology distribution in I��guciems	18
Figure 3. A chart showing population changes in Riga since the 19th century (Rub��ns, 2004, p. 9)	8	Figure 12. Industrial building typology distribution in I��guciems	18
Figure 4. A comparison chart showing average "old" apartment prices compared to prices in new residential projects in the years 2024 and 2025 (Valsts zemes dienests, 2025)	11	Figure 13. An early experimental variation of precast concrete residential buildings in I��guciems	19
Figure 5. Different ad-hoc loggia solutions in 1-464A series buildings in I��guciems	12	Figure 14. Different variations of the 1-464A series residential buildings in I��guciems	19
Figure 6. Location of I��guciems within Riga	14	Figure 15. Road network in I��guciems	20
Figure 7. Different series of Soviet period precast concrete panel residential buildings in I��guciems	15	Figure 16. A variation of the 1-464A series residential building in I��guciems	20
Figure 8. Residential building typology distribution in I��guciems	16	Figure 17. The comprehensive urban situation and zoning in I��guciems	21
Figure 9. Civic building typology distribution in I��guciems	16	Figure 18. The World Trade Centre building in Riga	26
		Figure 19. The KGB building in Riga, source (KGB Building, n.d.)	26

Figure 20. A new-built Soviet neighbourhood in Riga (Retro Latvia Club, n.d.)	28	Figure 39. A scheme showing the principle of the dependency script with all structural hierarchies and element relationship type inputs enabled	64
Figure 21. A Soviet neighbourhood today (I��guciems, Riga)	28	Figure 40. The principle of the partial removal script for exterior wall panels	66
Figure 22. A new construction currently ongoing in I��guciems, with series 1-464A residential buildings in the background	32	Figure 41. The principle of the partial removal script for interior wall panels	67
Figure 23. Original floor plans of the object building of the 1-464A series	39	Figure 42. The principle of the replacement script for interior wall panels	68
Figure 24. Cramped interior spaces typical of a 1-464A series residential building	42	Figure 43. The principle of the replacement script for exterior wall panels	69
Figure 25. Element catalogue in the context of the overall building assembly	46	Figure 44. Scenario A application example on a partial building floor	71
Figure 26. Individual element catalogue detailing unique precast components	47	Figure 45. Scenario B application example on a partial building floor	72
Figure 27. Inter-element relationship examples highlighting structural dependencies	50	Figure 46. Scenario C application example on a partial building floor	73
Figure 28. Original building assembly sequence	51	Figure 47. Scenario D application example on a partial building floor	74
Figure 29. Extracted dependency matrix mapping structural relationships	53	Figure 48. Elements for Scenario D with defined unmodified, locked and variable states	76
Figure 30. A computational workflow diagram for Option 1	55	Figure 49. The principle of the image sampler process	78
Figure 31. Comparative diagram illustrating architectural need versus structural reality in top-down floor plan alterations	56	Figure 50. The principle of the recursive dependency script	80
Figure 32. Reverse-triangle load distribution principle in a building section	56	Figure 51. The principle of sorting the geometries into containers	80
Figure 33. A computational workflow diagram for Option 2	57	Figure 52. The principle of different dependency script distinctions	81
Figure 34. An illustration showing possible structural collapse consequences if the dependency script is bypassed	58	Figure 53. The principle of the script for one whole floor iteration	81
Figure 35. A computational workflow diagram for Option 3	59	Figure 54. The image applied to the image sampler on the original input geometry	85
Figure 36. An illustration of the possible alteration process if Option 3 is developed further	60	Figure 55. The image applied to the image sampler on the 2nd floor	85
Figure 37. An illustration showing the spatial concerns of whole module removal	60	Figure 56. The adjusted image after the dependency script is triggered and forced scenarios are applied	85
Figure 38. A scheme showing the principle of the dependency script with varied structural hierarchy and element relationship type inputs	64	Figure 57. Experiment 1, part 1	86
		Figure 58. Experiment 1, part 2	87
		Figure 59. Experiment 1, part 3	89

Figure 60. Original applied gradients next to gradients adjusted by forced scenario application	91
Figure 61. Experiment 2, part 1: full height scenario application testing	91
Figure 62. Experiment 2, part 2: exoskeleton calculation and optional geometry removal	92
Figure 63. Scenario A spatial and programmatic evaluation	93
Figure 64. Scenario B spatial and programmatic evaluation	95
Figure 65. Scenario C spatial and programmatic evaluation	97
Figure 66. Scenario D spatial and programmatic evaluation	99
Figure 67. The original inputs of the image sampler (left) and the images adjusted through the application of forced scenarios (right) for Experiment 3	101
Figure 68. Programmatic diagrams of all the floors of the building for Experiment 3	102
Figure 69. Script generated geometries for floors 1-4 in Experiment 3	103
Figure 70. Script generated geometries for floor 5 and final script output geometry in Experiment 3	105
Figure 71. Floor plan outputs from the script generated geometry	106
Figure 72. An example of the post-algorithmic alteration output for Experiment 3	107
Figure 73. A section of the altered geometry output from Experiment 3	108
Figure 74. An example of the new-developed interior space from Experiment 3	108
Figure 75. The original inputs of the image sampler (left), the manually modified inputs (middle) and the images adjusted through the application of forced scenarios (right) for Experiment 4	109
Figure 76. Programmatic diagrams of all the floors of the building for Experiment 4	110
Figure 77. Script application trials and iterations for the ground floor in Experiment 4	111
Figure 78. Script application for the 2nd floor in Experiment 4	113

Figure 79. Output geometry from the algorithm with minor manual adjustments in Experiment 4	114
Figure 80. The original inputs of the image sampler (left) and the images adjusted through the application of forced scenarios (right) for Experiment 5	115
Figure 81. Algorithm application on the ground floor elements for Experiment 5	116
Figure 82. Algorithm application on the 2nd and 3rd floor elements for Experiment 5	117
Figure 83. Algorithm application on the 4th and 5th floor elements for Experiment 5	119
Figure 84. Original functional diagrams for all building floors (left) and adjusted functional diagrams after the floor plan extraction from the algorithmically altered building (right) for Experiment 5	121
Figure 85. Isometric view of removed elements in Experiment 5, applicable for reuse in further architectural synthesis	123
Figure 86. Building ground floor plan after architectural alterations and adjustments	131
Figure 87. Building 2nd floor plan after architectural alterations and adjustments	133
Figure 88. Building 3rd floor plan after architectural alterations and adjustments	135
Figure 89. Building 4th floor plan after architectural alterations and adjustments	137
Figure 90. Building 5th floor plan after architectural alterations and adjustments	139
Figure 91. Building cross-sections after architectural alterations and adjustments	141
Figure 92. Building longitudinal section after architectural alterations and adjustments	143
Figure 93. Building West and North elevations after architectural alterations and adjustments	145
Figure 94. Building East and South elevations after architectural alterations and adjustments	147
Figure 95. An interior view of a two-storey apartment	149
Figure 96. An interior view of the co-working space	149
Figure 97. An interior view of the two-storey gym	150

Figure 98. An exterior view of the exoskeleton-based terrace	150
Figure 99. An isometric diagram of distinct unmodified, modified, added and repurposed elements	151
Figure 100. Site plan illustrating the building within the reorganised community landscape	153
Figure 101. Images of the physical model showcasing a section of the transformed building	155
Figure 102. Images of the physical model showcasing a section of the transformed building - exterior	157
Figure 103. Images of the physical model showcasing a section of the transformed building - interior	159
Table 1. Comparison of different residential building construction periods in Riga and their key spatial characteristics	10
Table 2. Structural elements of the series 1-464A, their material composition, main issues and causes, and created model designation	35
Table 3. Room area comparison between the apartment types in the building	38
Table 4. Relationship types between building assembly elements	49
Table 5. Structural hierarchy between building assembly elements	49
Table 6. Applicability of created scenarios for each element type	65
Table 7. Scenario assignment to each element type	70
Table 8. A matrix defining unmodified, variable and locked state for each element type within the scenarios	75
Table 9. A scale to determine the grey value assignment to each scenario	77

12. BIBLIOGRAPHY

Aventino, M. (n.d.). Projekts Ilguciema mājas. Pillar Jaunie Projekti - jauni dzīvokļi Rīgā. Retrieved 2 May 2026, from <https://pillarliving.lv/lv/jaunie-projekti/ilguciema-majas/par-projektu>

Boverket. (2019). Boverket's building regulations – mandatory provisions and general recommendations, BBR (BFS 2011:6 with amendments up to BFS 2018:4). The Swedish National Board of Housing, Building and Planning. <https://www.boverket.se/en/start/publications/2019/boverkets-building-regulations--mandatory-provisions-and-general-recommendations-bbr/>

Clifford, B., McGee, W., & Muhonen, M. (2018). Recovering Cannibalism in Architecture with a Return to Cyclopean Masonry. *Nexus Network Journal*, 20(3), 583–604. <https://doi.org/10.1007/s00004-018-0392-x>

Department for Communities and Local Government. (2015). Technical housing standards – nationally described space standard. UK Government.

Dzirkalis, A., Lejiņš, K., Krivens, D., Dalka, A., Pilsētnieks, J., Brencis, R., Strazdiņš, M., Šnore, K., Lementujevs, A., Jonass, A., & Dzudziļo, M. (2019). Ziņojums '464. Sērijas daudzdzīvokļu dzīvojamās ēkas' Rīga, Jelgava (No. 2019/58). SIA 'CMB' (for Ekonomikas ministrija).

Easterling, K. (2014). *Subtraction*. Sternberg Press.

Eschenbach, M. B., Wagner, A.-K., Ledderose, L., Böhret, T., Wohlfeld, D., Gille-Sepehri, M., Kuhn, C., Kloft, H., & Tessmann, O. (2023). Matter as Met: Towards a Computational Workflow for Architectural Design with Reused Concrete Components. In C. Gengnagel, O. Baverel, G. Betti, M. Popescu, M. R. Thomsen, & J. Wurm (Eds), *Towards Radical Regeneration* (pp. 442–455). Springer International Publishing. https://doi.org/10.1007/978-3-031-13249-0_35

Groskaufmanis, M., & Ozola, E. (Eds). (2019). *Kopā un atsevišķi: Daudzdzīvokļu namu arhitektūra Latvijas ekonomiskajā, politiskajā un sociālajā ainavā. FOLD sadarbībā ar Latvijas Jaunā teātra institūtu.*

KGB Building. (n.d.). Retrieved 30 April 2026, from <https://okupacijasmuzejs.lv/en>

Kloft, H., Baghdadi, A., Ledderose, L., Eschenbach, M. B., Tessmann, O., Kuhn, C., Wagner, A.-K., Ayres, P., Thomsen, M. R., Sheil, B., & Skavara, M. (2024). TOWARDS A HYBRID MODULAR CONSTRUCTION SYSTEM REUSING PRECAST CONCRETE COMPONENTS. In *Fabricate 2024: Creating Resourceful Futures* (pp. 106–113). UCL Press. <http://www.jstor.org/stable/jj.11374766.17>

Košovojs, E. (2014). Русский: 'Угловой дом' в РигеLatviešu: Stūra māja [Graphic]. https://lv.wikipedia.org/wiki/Att%C4%93ls:St%C5%ABra_m%C4%81ja.JPG. https://commons.wikimedia.org/wiki/File:St%C5%ABra_m%C4%81ja.JPG

Latvijas Sabiedriskie Mediji (LSM). (2020, June 3). Atbalstu koncertzāles būvniecībai Elizabetes ielā 2 pauž arī LNSO un domubiedri. <https://www.lsm.lv/raksts/kultura/kulturtelpa/atbalstu-koncertzales-buvniecibai-elizabetes-iela-2-pauz-ari-lnso-un-domubiedri.a365914/>

Latvijas Sabiedriskie Mediji (LSM). (2023, October 20). Ēka Elizabetes ielā 2 no «komunisma rēga» pārtop par «Brīvības teritoriju». Latvijas Sabiedriskie Mediji (LSM). <https://www.lsm.lv/raksts/kultura/kulturtelpa/20.10.2023-eka-elizabetes-iela-2-no-komunisma-rega-partop-par-brivibas-teritoriju.a528542/>

Latvijas Sabiedriskie Mediji (LSM). (2024, March 26). Valsts mēģinās pārdot Stūra māju; Okupācijas muzeja rīcībā telpas būs vismaz līdz 2050. Gadam. Latvijas Sabiedriskie Mediji (LSM). <https://www.lsm.lv/raksts/kultura/kulturtelpa/26.03.2024-valsts-meginas-pardot-stura-maju-okupacijas-muzeja-riciba-telpas-bus-vismaz-lidz-2050-gadam.a548145/>

Lejiņš, K., Pilsētnieks, J., & Brencis, R. (2019). Tehniskās apsekošanas atzinums: Daudzdzīvokļu dzīvojamā ēka, Kurzemes prospekts 56, Rīga (No. 2019/58). SIA 'CMB' (for Ekonomikas ministrija).

Meuser, P., Zadorin, D., & Zadorin, D. (2015). Towards a typology of Soviet mass housing: Prefabrication in the USSR 1955 – 1991. DOM publ.

nbhd. (2023, August 19). *Loved, Hated, Misunderstood: Microrayons of Riga. Neighborhood.* <https://neighborhood.lv/en/real-estate/loved-hated-misunderstood-microrayons-riga/>

Önalán, B., Mitropoulou, I., Triantafyllidis, E., Hunhevicz, J., & De Wolf, C. (2025). Computational methods for circular design with non-standard materials: Systematic review and future directions. *International Journal of Architectural Computing*. <https://doi.org/10.1177/14780771251316125>

Retro Latvia Club. (n.d.).

Rubīns, J. (2004). Rīgas dzīvojamais fonds 20 gadsimtā: Tipoloģiskā rakursā. Jumava.

Sarnovičs, J. (2026). Ilguciems. In *Nacionālā enciklopēdija*. <https://enciklopedija.lv/skirklis/281809>

Šerman, K., & Horvat, J. (2023). On the Phenomenon of Decay in Architecture: Theoretical Perspectives from the 1950s to the 2020s. *Prostor*, 31(2(66)), 168–181. [https://doi.org/10.31522/p.31.2\(66\).3](https://doi.org/10.31522/p.31.2(66).3)

Strazdiņš, M., Šnore, K., & Lementujevs, A. (2019). Nesošo konstrukciju un mezglu stiprības novērtējums '464. Sērijas daudzdzīvokļu dzīvojamās ēkas' Rīga, Jelgava (No. 2019/58). SIA 'CMB' (for Ekonomikas ministrija).

Treija, S., & Bratuškins, U. (2019). Socialist Ideals and Physical Reality: Large Housing Estates in Riga, Latvia (pp. 161–180). https://doi.org/10.1007/978-3-030-23392-1_8

Tzourmakliotou, D. (2021). Designing for Deconstruction—The Related Factors. *Journal of Civil Engineering and Architecture*, 15(9). <https://doi.org/10.17265/1934-7359/2021.09.002>

Valsts zemes dienests. (2025). *Nekustamā īpašuma tirgus datu bāzes atvērtie dati* (Version 2025. gadā NĪTIS reģistrētie darījumi ar NĪ) [Data set]. Latvijas Atvērtā datu portāls. <https://data.gov.lv/dati/lv/dataset/nekustama-ipasuma-tirgus-datu-bazes-atvertie-dati>

Zariņa, Z., & Uzulēns, J. (2023). Technical Condition of Soviet-Era Apartment Buildings, Related Problems and Possible Solutions in Latvia. *Baltic Journal of Real Estate Economics and Construction Management*, 11(1), 257–269. <https://doi.org/10.2478/bjreecm-2023-0016>

Zboinska, M. A., & Göbel, F. (2025). Digital tool integrations for architectural reuse of salvaged building materials. *Automation in Construction*, 170. <https://doi.org/10.1016/j.autcon.2024.105947>