

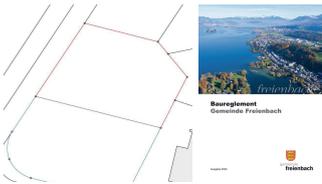
**Bachelor of Arts in Digital Construction**

**Digital unterstützte Machbarkeitsstudie**

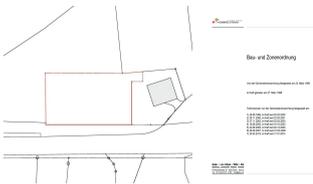
Prozess der Machbarkeitsstudie



M 1:750 / Reglement Freienbach



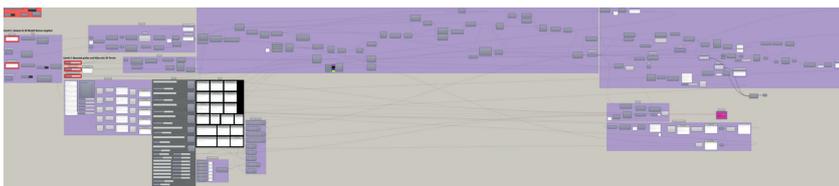
M 1:750 / Reglement Hombrechtikon



M 1:750 / Reglement Lachen



Grasshopper Skript

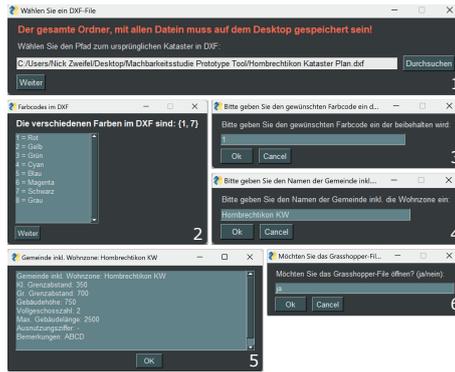


**Problemstellung**

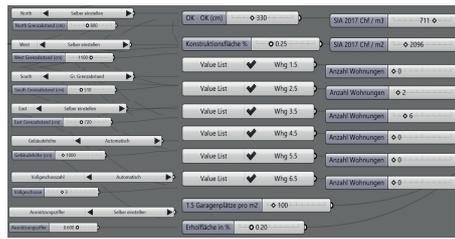
Machbarkeitsstudien in der Bauindustrie sind komplexe Prozesse, die viel Ressourcen in Anspruch nehmen. Die steigenden Kosten für Bauland erfordern präzisere Machbarkeitsstudien zur Maximierung der Parzellenausnutzung. Der stetige Fortschritt der Digitalisierung bietet ein enormes Potenzial, diese Prozesse durch den Einsatz digitaler Tools und Automation zu optimieren. Dabei stellen vor allem die Vielfältigkeit der Projekte und heterogene Strukturierung von Reglementen und Normen der verschiedenen Gemeinden eine Herausforderung dar.

Angesichts der Problemstellung war das Ziel dieser Arbeit die Untersuchung der Möglichkeiten und Grenzen einer Automatisierung der technischen Machbarkeit mittels digitaler Technologien. Insbesondere wurde ein Prototyp zur Optimierung von solcher Studien entwickelt. Methodisch wurde eine Analyse bestehender digitaler Tools durchgeführt. Darauf aufbauend erfolgte dann die Entwicklung eines digitalen Prototyps, der auf Python und Grasshopper basiert.

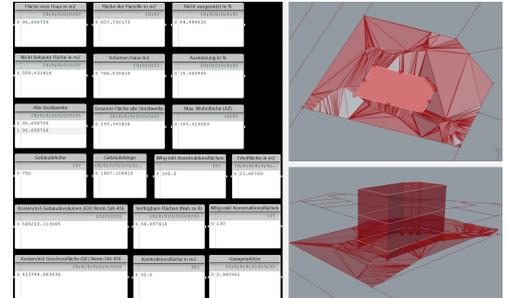
Python Anwendung für Aufarbeitung



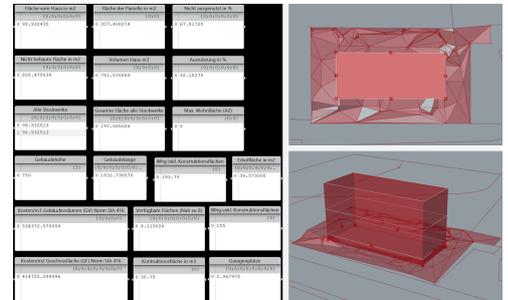
Grasshopper 3D Steuerung



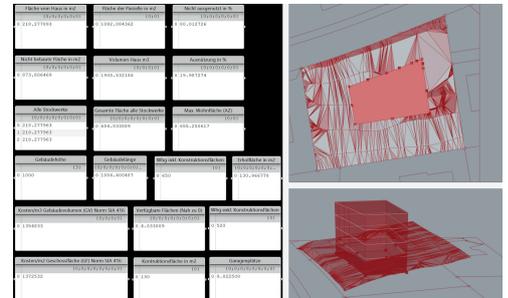
Technische Machbarkeit Parzelle Freienbach



Technische Machbarkeit Parzelle Hombrechtikon



Technische Machbarkeit Parzelle Lachen



**Lösungskonzept**

Es wurden verschiedene Skripte entwickelt, um automatisiert 3D-Modelle zu erstellen und die erlaubten Bauten auf einer Parzelle zu visualisieren. Die Ergebnisse zeigen, dass der entwickelte Prototyp das Potenzial hat, die Effizienz von Machbarkeitsstudien erheblich zu steigern. Trotz einiger Herausforderungen, wie der Deutung von Baureglementen und dem Umgang mit komplexen Grundstücken, konnte das Tool erfolgreich getestet und angewendet werden. Es wurde ein erhebliches Zeitersparnis im Vergleich zur herkömmlichen Durchführung von Machbarkeitsstudien erreicht. Auch wurde erkannt, dass wenn die technische Machbarkeit automatisiert werden kann, besteht die Chance, die übrigen Überprüfungen zu verknüpfen. Dies ermöglicht eine schnelle Nachprüfung weiterführender Varianten, insbesondere der wirtschaftliche Aspekt.

Für zukünftige Forschungen wird empfohlen, das entwickelte Tool weiter zu optimieren. Es sollten Aspekte wie Energieeffizienz, Ressourcenverbrauch und Rückbaukosten integriert werden. Zudem wird eine Verbesserung der Benutzerfreundlichkeit und

eine stärkere Integration mit anderen digitalen Datenquellen angestrebt. Eine Langzeitstudie zur Anwendung des Tools in der Praxis könnte wertvolle Erkenntnisse liefern.

**Nick Zweifel**

Betreuer:  
Frank Thesseling  
Michal Rontsinsky

Experte:  
Joanna Demkow

Dozenten:  
Markus Weber  
Mark Baldwin

# Hochschule Luzern

Technik & Architektur

---

## Fachartikel zur Bachelor-Thesis

Im Studiengang

«Digital Construction in Architecture»

# Digital unterstützte Machbarkeitsstudie

Verfasst von: **Nick Zweifel**

Matrikelnummer: 20-275-434

E-Mail: [nick.zweifel@stud.hslu.ch](mailto:nick.zweifel@stud.hslu.ch) / [nickzweifel@bluewin.ch](mailto:nickzweifel@bluewin.ch)

Erstbetreuer: Frank Thesseling

Zweitbetreuer: Michal Rontsinsky

Experte: Joanna Demkow

Dozenten: Markus Weber, Mark Baldwin

Abgabedatum: 09.06.2023

**Hinweis:** Dieser Fachartikel ist Teil der Bachelor-Thesis und wurde von keinem Dozenten nachbearbeitet. Veröffentlichungen (auch auszugsweise) sind ohne das Einverständnis des Instituts Gebäudetechnik und Energie der Hochschule Luzern – Technik & Architektur nicht erlaubt.

# Digital unterstützte Machbarkeitsstudie

Die Durchführung von Machbarkeitsstudien in der Bauindustrie ist ein komplexer Prozess, der viel Zeit und Ressourcen beansprucht. Der Baulandpreis steigt, was immer genauere Machbarkeitsstudien verlangt, um die Parzellenfläche maximal ausnutzen zu können. Der stetige Fortschritt der Digitalisierung bietet ein enormes Potenzial, diese Prozesse durch den Einsatz digitaler Tools und Automation zu optimieren. Dabei stellen insbesondere die Vielfältigkeit der Projekte und die heterogene Strukturierung von Baureglementen und Normen der verschiedenen Gemeinden eine Herausforderung dar. Das erschwert die Entwicklung einer universellen Lösung, die von allen gleichermassen genutzt werden kann.

Angesichts dieser Problemstellung war das Ziel dieser Arbeit die Untersuchung der Möglichkeiten und Grenzen einer gewissen Automatisierung von Machbarkeitsstudien mittels digitaler Technologien. Im Speziellen wurde die Entwicklung eines Prototyps angestrebt, der in der Lage ist, die Erstellung von Machbarkeitsstudien zu optimieren und beschleunigen. Es wurden diverse Skripte entwickelt, um automatisch 3D-Modelle zu erstellen und die erlaubten Bauten auf einer Parzelle visuell darzustellen.



Abb. 1: Verlauf einer Machbarkeitsstudie

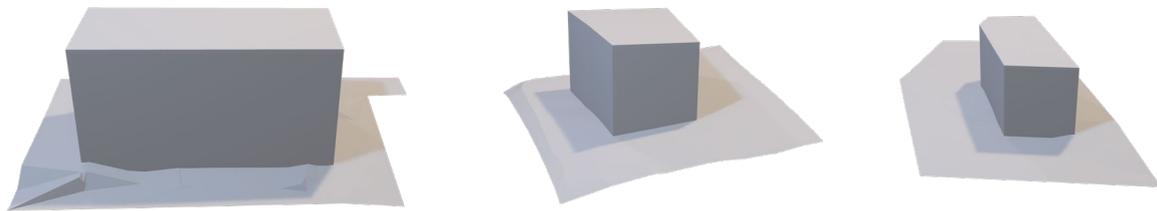


Abb. 2: STL 3D-Dateien (Hombrechtikon, Freienbach & Lachen)

## Methodik

Die Methodik der Bachelorarbeit umfasst vier Hauptphasen: Recherche und Analyse, Entwicklung eines Prototyps, Anwendung und Fazit. In der ersten Phase werden relevante Reglemente, Artikel und Berichte analysiert, um ein Verständnis für Machbarkeitsstudien und Tools zu erlangen. Die Analyse konzentriert sich auf bestehende Tools wie Spacemaker, Archistar und Delve, um Anwendungsbereiche, Vor- und Nachteile darzustellen.

Insbesondere wird untersucht, wie Python und Grasshopper für den Prototyp eingesetzt werden können. Die zweite Phase zielt auf die Entwicklung einer digital unterstützten technischen Machbarkeitsstudie ab, wobei eigene Skripte die Effizienz und Genauigkeit optimieren sollen. Die dritte Phase zeigt einen Workflow auf, basierend auf dem Prototyp und Erkenntnissen aus der zweiten Phase. Hier werden erfolgreiche Beispiele sowie Herausforderungen und Grenzen präsentiert. In der letzten Phase werden Ergebnisse zusammengefasst, die Forschungsfrage diskutiert und Empfehlungen für die Zukunft gegeben.

## Ergebnisse

Die Effizienz des Tools wird durch die Strukturierung und Interpretationsfreiheit der relevanten Baureglemente beeinflusst, was in Bezug auf die Gemeinden Hombrechtikon, Lachen und Freienbach gezeigt wird. Eine Prüfung der technischen Machbarkeit, die im traditionellen Kontext bis zu 3 Wochen dauern kann, wird durch den Prototyp auf eine Stunde reduziert, ein Zeitersparnis von 98 %. Dies ermöglicht eine schnellere Grundlage für weitere Wirtschafts- und Gesellschaftsanalysen. Es ist jedoch wichtig zu beachten, dass dieses Zeitersparnis nicht eine vollständige Machbarkeitsstudie darstellt, sondern vielmehr ein Teil des gesamten Planungsablaufs ist. Änderungen und Varianten können schnell vorgenommen und überprüft werden, da die finanziellen Aspekte an die technischen Vorgaben geknüpft sind.

Der Prototyp hat auch gewisse Einschränkungen. Er kann Schwierigkeiten bei der Erfassung komplexer Parzellenformen haben, namentlich solcher mit sieben oder mehr Seiten. Zukünftige Versionen könnten dieses Problem durch einen Algorithmus beheben, der komplexe Parzellenformen effizient erkennt. Aktuell fehlt dem Prototyp auch ein Algorithmus zur Gebäudeteilung basierend auf einer bestimmten Länge, was seine Anwendbarkeit für längere Gebäude einschränkt. Die Implementierung eines solchen Algorithmus würde seine Fähigkeiten erweitern und die Planung von grösseren Arealen mit mehreren Gebäuden ermöglichen. Trotz dieser Limitationen ist der Prototyp ein nützliches Werkzeug, das durch weitere Verbesserungen noch effektiver gemacht werden kann.

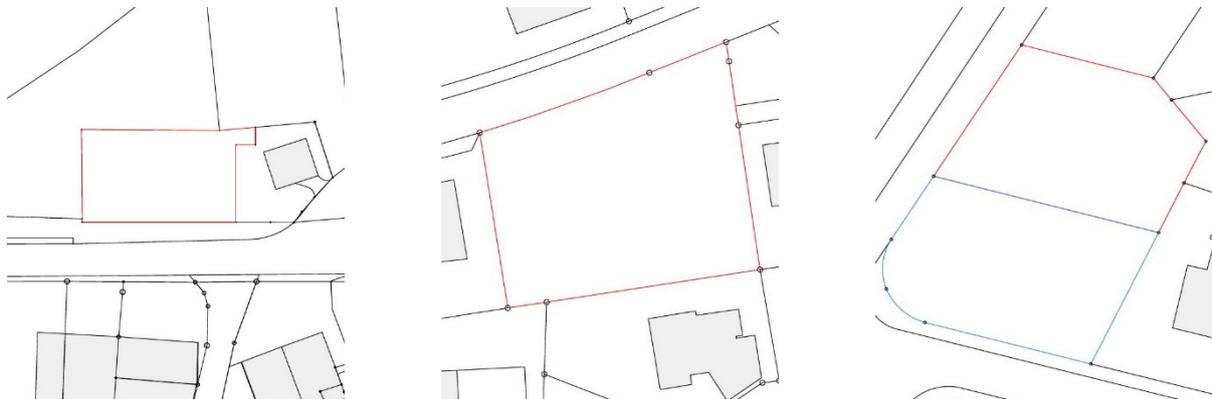


Abb. 3: Katasterpläne im Massstab 1:1000 (Hombrechtikon, Freienbach & Lachen)

Über eine im Rahmen des Projekts geschriebene Python Anwendung, gelangen Katasterpläne, 3D-Terrain und Baureglement Informationen in ein Grasshopperskript. Die Baureglement-Informationen wurden zuvor in ein Excel geschrieben und beinhalten Angaben zur Gemeinde inkl. Wohnzone, Kl. Grenzabstand, Gr. Grenzabstand, Gebäudehöhe, Vollgeschosszahl, Max. Gebäudelänge und Ausnutzungsziffer. Diese werden in die Grasshopper-Datei automatisch aufgenommen und in der 3D-Ansicht visualisiert. Der Benutzer hat die Möglichkeit Änderungen vorzunehmen und erhält sofort Feedback in einem Output-Panel, in Form von Zahlen, die das korrespondierende 3D-Modell beschreiben. Die Bearbeitung kann mit Slider vernommen werden und beziehen sich auf die Parzellengrenzen und Gebäudehöhe. Anhand der Grösse vom Gebäude können anschliessend Aussagen über Kosten, Garagenplätze, zu bauende Erholungsflächen und Wohnungseinheiten gemacht werden.



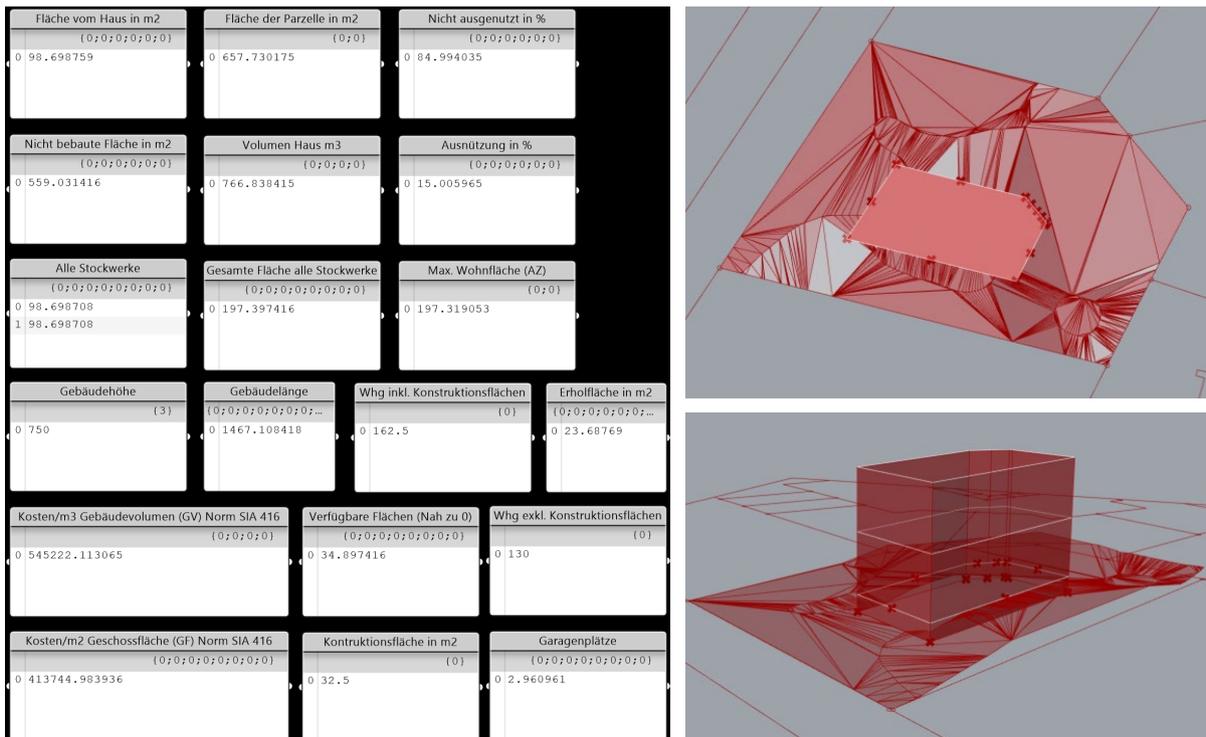


Abb. 6: Technische Machbarkeit Lachen

## Diskussion

Die Arbeit erforschte digitale Möglichkeiten zur Effizienzsteigerung in Machbarkeitsstudien und bestätigte ihre Forschungsthese durch einen Prototyp. Sie deckte verschiedene Aspekte von Machbarkeitsstudien und digitalen Technologien ab, analysierte bestehende Tools und zeigte ihre Entwicklung. Es wurde erkannt, dass die Automatisierung erhebliches Potenzial birgt, wobei der Erfolg stark von der Datengrundlage abhängt. Die Anwendung des Prototyps ergab erhebliche Zeitersparnisse von bis zu 98 % in der Überprüfung der technischen Machbarkeit und eine schnellere Entscheidungsfindung für weitere Schritte in der Machbarkeitsstudie. Dennoch ist der Prototyp erst der Anfang eines langen Weges zur Optimierung von Machbarkeitsstudien, da baurechtliche Normen und Vorschriften strukturiert werden müssen, was seitens der jeweiligen Gemeinden geschehen muss.

Zukünftige Forschungen sollten auf die Verbesserung der Benutzerfreundlichkeit des Tools abzielen, seine Anpassungsfähigkeit erhöhen und die Integration mit anderen Anwendungen fördern. Das beinhaltet eine GIS-Anbindung und automatische Aktualisierung der Daten. Ebenso sollte eine breitere Palette von Aspekten wie Energieeffizienz und Rückbau bestehender Gebäude berücksichtigt werden. Langzeitstudien und Fallstudien könnten helfen, Benutzerfeedback zu sammeln und Verbesserungspotenziale zu identifizieren. Trotz der Skepsis gegenüber der Automation und deren Auswirkung über das menschliche Urteilsvermögen bestätigt die Arbeit das Potenzial digitaler Tools zur Effizienzsteigerung.

# Abstract English

Conducting feasibility studies for buildings is a process that takes a lot of time and money. Building land is also becoming more expensive, which calls for increasingly accurate feasibility studies to utilize the plot area as efficiently as possible. The steady progress of digitalization offers great potential to optimize this process using digital tools. In this context, the diversity of projects and the heterogeneous structuring of building regulations and standards of the different municipalities in Switzerland pose a challenge.

This work aimed to investigate the possibilities and limitations of automation in feasibility studies using digital technologies, primarily the technical feasibility. The focus was the development of a prototype that optimizes and accelerates the technical feasibility assessment process. Methodologically, an analysis of existing digital tools was conducted. Based on this, a digital prototype based on Python and Grasshopper was developed. Different scripts were developed to automatically create 3D models and visualize the allowed buildings on a parcel. The results show that the developed prototype has the potential to significantly increase the efficiency of feasibility studies. The time to verify the technical feasibility could be reduced by up to 98 %. Despite some challenges, such as interpreting building regulations and handling complex land, the tool was successfully tested and applied. A noticeable time saving was achieved compared to the traditional way of conducting feasibility studies.

For future research, it is recommended to further optimize the developed tool. Aspects such as energy efficiency, resource consumption, and deconstruction costs should be integrated. In addition, an improvement in usability and a more dynamic integration with other digital data sources is sought. A long-term study on the application of the tool in practice could provide valuable insights.

# Hochschule Luzern

Technik & Architektur

---

## Bachelorarbeit

Digital unterstützte Machbarkeitsstudie

---

Verfasst von:

**Nick Zweifel**

Matrikelnummer: 20-275-434

E-Mail: [nick.zweifel@stud.hslu.ch](mailto:nick.zweifel@stud.hslu.ch) / [nickzweifel@bluewin.ch](mailto:nickzweifel@bluewin.ch)

Erstbetreuer: Frank Thesseling

Zweitbetreuer: Michal Rontsinsky

Experte: Joanna Demkow

Dozenten: Markus Weber, Mark Baldwin

Abgabedatum: 09. Juni 2023

# **Bachelor-Thesis an der Hochschule Luzern - Technik & Architektur**

**Titel** **Digital unterstützte Machbarkeitsstudie**

**Diplomandin/Diplomand** **Zweifel, Nick**

**Bachelor-Studiengang** **Bachelor Digital Construction**

**Semester** **FS23**

**Dozentin/Dozent** **Thesseling, Frank / Rontsinsky, Michal**

**Expertin/Experte** **Demkow, Joanna**

## **Abstract Deutsch**

Die Durchführung von Machbarkeitsstudien im Bauwesen ist ein Prozess, der viel Zeit und Geld kostet. Auch das Bauland wird teurer, was immer genauere Machbarkeitsstudien fordert, um die Parzellenfläche so effizient wie möglich ausnutzen zu können. Der stetige Fortschritt der Digitalisierung bietet ein grosses Potenzial, diesen Prozess durch den Einsatz digitaler Tools zu optimieren. Dabei stellen insbesondere die Vielfältigkeit der Projekte und die heterogene Strukturierung von Baureglementen und Normen der verschiedenen Gemeinden eine Herausforderung dar.

Ziel dieser Arbeit war die Untersuchung der Möglichkeiten und Grenzen einer Automatisierung in Machbarkeitsstudien mittels digitaler Technologien, primär die technische Machbarkeit. Die Entwicklung eines Prototyps, der den Prozess der technischen Machbarkeitsbewertung optimiert und beschleunigt, stand im Fokus. Methodisch wurde eine Analyse bestehender digitaler Tools durchgeführt. Darauf aufbauend erfolgte die Entwicklung eines digitalen Prototyps, der auf Python und Grasshopper basiert. Es wurden verschiedene Skripte entwickelt, um automatisiert 3D-Modelle zu erstellen und die erlaubten Bauten auf einer Parzelle zu visualisieren. Die Ergebnisse zeigen, dass der entwickelte Prototyp das Potenzial hat, die Effizienz von Machbarkeitsstudien erheblich zu steigern. Die Zeit um die technische Machbarkeit überprüfen konnte bis zu 98 % reduziert werden. Trotz einiger Herausforderungen, wie der Deutung von Baureglementen und dem Umgang mit komplexen Grundstücken, konnte das Tool erfolgreich getestet und angewendet werden. Es wurde ein spürbares Zeitersparnis im Vergleich zur herkömmlichen Durchführung von Machbarkeitsstudien erreicht.

Für zukünftige Forschungen wird empfohlen, das entwickelte Tool weiter zu optimieren. Es sollten Aspekte wie Energieeffizienz, Ressourcenverbrauch und Rückbaukosten integriert werden. Zudem wird eine Verbesserung der Benutzerfreundlichkeit und eine dynamischere Integration mit anderen digitalen Datenquellen angestrebt. Eine Langzeitstudie zur Anwendung des Tools in der Praxis könnte wertvolle Erkenntnisse liefern.

Ort, Datum **Rapperswil, 09. Juni 2023**

© **Nick Zweifel, Hochschule Luzern – Technik & Architektur**

---

Alle Rechte vorbehalten. Die Arbeit oder Teile davon dürfen ohne schriftliche Genehmigung der Rechteinhaber weder in irgendeiner Form reproduziert noch elektronisch gespeichert, verarbeitet, vervielfältigt oder verbreitet werden.

Sofern die Arbeit auf der Website der Hochschule Luzern online veröffentlicht wird, können abweichende Nutzungsbedingungen unter Creative-Commons-Lizenzen gelten. Massgebend ist in diesem Fall die auf der Website angezeigte Creative-Commons-Lizenz.

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis .....</b>	<b>5</b>
<b>1 Einleitung .....</b>	<b>7</b>
1.1 Hintergrund und Motivation.....	7
1.2 Zielsetzung der Arbeit.....	7
1.3 Forschungsthese .....	8
1.4 Methodik und Vorgehensweise .....	8
<b>2 Grundlagen einer Machbarkeitsstudie.....</b>	<b>9</b>
2.1 Definition und Zweck.....	9
2.2 Prozess einer Machbarkeitsstudie .....	9
2.3 Zuvor erhobene Daten.....	10
2.4 Datenerhebung aus der Machbarkeitsstudie .....	11
2.5 Zusammenwirkung der Daten .....	12
2.6 Abgrenzung der Automation in einer Machbarkeitsstudie.....	12
<b>3 Effizienz in der Bauindustrie.....</b>	<b>13</b>
3.1 Definition von Effizienz .....	13
3.2 Digitale Effizienz in vier Bereiche.....	13
<b>4 Technologien auf dem Markt .....</b>	<b>14</b>
4.1 Bestehende Tools .....	14
4.1.1 Sidewalk Labs .....	14
4.1.2 Spacemaker .....	15
4.1.3 Archistar .....	16
4.2 Erkenntnisse aus den Tools.....	16
<b>5 Entwicklung der digital unterstützten Machbarkeitsstudie.....</b>	<b>17</b>
5.1 Auswahl und Begründung der eingesetzten Technologien .....	17
5.2 Datenintegration und Aufbereitung .....	17
5.2.1 Baureglement.....	17
5.2.2 Katasterplan .....	19
5.2.3 Gewachsenes Terrain .....	20
5.3 Automatisierte Zusammenführung.....	20
5.4 Erstellung von 3D-Modellen .....	22
5.5 Skriptbeschreibung und Funktionsweise .....	22
5.5.1 Python: Baureglement Search Code.....	22

---

5.5.2 Python: Katasterplan-Analyse und Baureglement-Verarbeitungstool .....	24
5.5.3 Python: Eingabe Tool Baureglementdaten.....	28
5.5.4 Grasshopper: 3D-Modell Erstellung .....	30
<b>6 Anwendung .....</b>	<b>34</b>
6.1 Beschreibung des Anwendungsfalls.....	34
6.2 Datenintegration und Aufbereitung .....	34
6.3 Zusammenführung.....	36
6.4 Erstellung von 3D-Modellen .....	37
6.4.1 Lachen W2.....	40
6.4.2 Freienbach W3 .....	41
6.5 Implementierte Informationen im Output .....	42
6.6 Fazit der Erstellung .....	43
6.7 Abgrenzung.....	44
<b>7 Diskussion.....</b>	<b>45</b>
7.1 Zusammenfassung der Arbeit.....	45
7.2 Beantwortung der Forschungsfrage .....	45
7.3 Empfehlung für zukünftige Forschungen.....	46
<b>8 Reflexion .....</b>	<b>47</b>
<b>9 Literaturverzeichnis .....</b>	<b>48</b>

## Abbildungsverzeichnis

Abb. 1: Grafik Prozess (Eigene Darstellung) .....	10
Abb. 2: Digitale Transformation (Haustechnik Dialog, 2016) .....	13
Abb. 3: Interface Delve (Gibson, 2020) .....	14
Abb. 4: Interface Spacemaker (Frausto-Robledo, 2020) .....	15
Abb. 5: Interface Archistar (Archistar, n.d.).....	16
Abb. 6: Baureglement Search Fenster (Eigene Darstellung).....	18
Abb. 7: Baureglement Search Ausgabe (Eigene Darstellung).....	18
Abb. 8: Eingabefenster (Eigene Darstellung).....	19
Abb. 9: Baureglement Excel (Eigene Darstellung) .....	19
Abb. 10: Gemeinde Lachen Ausschnitt (Gemeinde Lachen, 1995) .....	21
Abb. 11: Gemeinde Hombrechtikon Ausschnitt (Gemeinde Hombrechtikon, 1998) .....	21
Abb. 12: Gemeinde Freienbach Ausschnitt (Gemeinde Freienbach, 2022) .....	21
Abb. 13: Übersicht Grasshopper Skript (Eigene Darstellung).....	30
Abb. 14: Input Gruppierung (Eigene Darstellung) .....	31
Abb. 15: Einstellungsslider (Eigene Darstellung) .....	31
Abb. 16: Katasterplan Verarbeitung (Eigene Darstellung).....	31
Abb. 17: 3D-Terrain Verarbeitung (Eigene Darstellung).....	31
Abb. 18: Excel Aufspaltung (Eigene Darstellung).....	32
Abb. 19: Parzelle filtern nach N, O, S, W für Parzellengrenzen (Eigene Darstellung) .....	32
Abb. 20: Terrain / Haus Verschnitt (Eigene Darstellung).....	32
Abb. 21: Stockwerk Bearbeitung / Galapagos (Eigene Darstellung).....	33
Abb. 22: Output Panel (Eigene Darstellung).....	33
Abb. 23: Berechnung (Eigene Darstellung).....	33
Abb. 24: Steuerung Slides (Eigene Darstellung) .....	33
Abb. 25: Berechnung (Eigene Darstellung).....	33
Abb. 26: Katasterplan Hombrechtikon (Eigene Darstellung).....	34
Abb. 27: Katasterplan Lachen (Eigene Darstellung) .....	35
Abb. 28: Katasterplan Freienbach (Eigene Darstellung).....	35
Abb. 29: Eingabe Bauvorgaben (Eigene Darstellung) .....	35
Abb. 30: Bauvorgaben Excel (Eigene Darstellung).....	35
Abb. 31: 3D-Terrain Modelle (Hombrechtikon, Lachen, Freienbach) (Eigene Darstellung) .....	36
Abb. 32: Prototyp Tool Ordner (Eigene Darstellung).....	36
Abb. 33: Schritt 1 der Anwendung (Eigene Darstellung) .....	36
Abb. 34: Verschiedenen Farben im DXF (Eigene Darstellung).....	37
Abb. 35: Farbe angeben (Eigene Darstellung) .....	37
Abb. 36: Gemeinde Abfrage (Eigene Darstellung).....	37
Abb. 37: Gemeindeinformationen Ausgabe (Eigene Darstellung).....	37
Abb. 38: Grasshopper Öffnung (Eigene Darstellung) .....	37
Abb. 39: Namen eingeben (Eigene Darstellung).....	38
Abb. 40: Z-Achse Slider (Eigene Darstellung) .....	38
Abb. 41: Massstab Änderung (Eigene Darstellung) .....	38
Abb. 42: Mittelpunkt verschieben (Eigene Darstellung).....	38

---

Abb. 43: Steuerung und korrespondierendes Modell (Eigene Darstellung).....	38
Abb. 44: Steuerung unterer Teil (Eigene Darstellung).....	39
Abb. 45: Modell Grundriss und Ansicht (Eigene Darstellung) .....	39
Abb. 46: STL-Modell Hombrechtikon (Eigene Darstellung) .....	39
Abb. 47: Output Panel (Eigene Darstellung).....	39
Abb. 48: Steuerung zur Machbarkeitslösung (Eigene Darstellung) .....	40
Abb. 49: Modell Machbarkeit Lachen (Eigene Darstellung) .....	40
Abb. 50: STL-Modell Lachen (Eigene Darstellung).....	40
Abb. 51: Output Panel Lachen (Eigene Darstellung) .....	40
Abb. 52: Steuerung zur Machbarkeitslösung (Eigene Darstellung) .....	41
Abb. 53: Modell Machbarkeit Freienbach (Eigene Darstellung).....	41
Abb. 54: STL-Modell Freienbach (Eigene Darstellung) .....	41
Abb. 55: Output Panel Freienbach (Eigene Darstellung).....	41
Abb. 56: Veranschaulichung der Slider (Eigene Darstellung) .....	43

# 1 Einleitung

## 1.1 Hintergrund und Motivation

Die zunehmende Urbanisierung und die Notwendigkeit einer effizienteren Projektentwicklung stellen Planer vor immer grössere Herausforderungen. Um ein geeignetes Projekt für eine Parzelle umsetzen zu können, bedarf es einer Machbarkeitsstudie. Diese bietet eine solide Ausgangslage für die Beurteilung von Parzellen und die frühe Identifizierung von Risiken bei der Entwicklung. Auch wird das Bauland immer teurer, im Kanton Zürich ist der Quadratmeterpreis 2007 bis 2021 von 1390.- auf 2270.- (+63%) gestiegen (Dreier, 2022). Es ist daher wichtig, die Fläche so effizient wie möglich zu nutzen, um den grösstmöglichen Gewinn aus der Parzelle herauszuholen.

Da diese Studien zeitaufwendig sind, macht es Sinn digitale Werkzeuge einzusetzen, um den Prozess effizienter und präziser durchzuführen. In den letzten Jahren hat das Interesse der Digitalisierung in vielen Bereichen der Architektur, Planung und Bauwesen zugenommen. Die Integration digitaler Technologien in den Prozess der Machbarkeitsstudie bietet die Möglichkeit, die Qualität und Genauigkeit von Projektauswertungen zu verbessern und gleichzeitig Ressourcen wie Geld und Zeit zu sparen. Obwohl bereits einige Untersuchungen zu digitalen Technologien in Machbarkeitsstudien durchgeführt wurden, sind im schweizerischen Kontext spezifische Herausforderungen aufgrund der unterschiedlichen Grundlagen und Vorgaben der Gemeinden zu berücksichtigen. Das erschwert die Entwicklung einer universellen Lösung, die von allen gleichermassen genutzt werden kann. Daher besteht weiterhin Bedarf, die Potenziale und Herausforderungen einer automatisierten Machbarkeitsstudie unter Berücksichtigung der regionalen Besonderheiten der Schweiz zu erforschen und zu verstehen.

Die Motivation dieser Bachelorarbeit liegt in der Erkenntnis, dass die Digitalisierung das Potenzial hat, den Prozess der Machbarkeitsstudie zu verändern, verbessern und effizienter zu gestalten. Durch die Untersuchung des Themas, der Anwendung und der damit verbundenen Herausforderungen soll diese Arbeit dazu beitragen, das Verständnis der Digitalisierung in der Machbarkeitsstudie zu erweitern und neue Perspektiven für zukünftige Anwendungen zu ermöglichen.

## 1.2 Zielsetzung der Arbeit

Ziel der Bachelorarbeit ist es, die Möglichkeiten und Herausforderungen der Digitalisierung bei Machbarkeitsstudien für Parzellen zu untersuchen. Dabei sollen Grundlagen und Prozesse einer Machbarkeitsstudie erläutert werden. Danach wird der aktuelle Stand der Technik und die praktische Umsetzung dieser Methoden anhand von Skripts analysiert. Daraus sollen Empfehlungen für die erfolgreiche Integration digitaler Technologien in Machbarkeitsstudien hervorgehen.

### 1.3 Forschungsthese

Digitale Tools und Technologien können den Prozess der Machbarkeitsstudie erheblich unterstützen, indem sie gezielte Effizienzsteigerungen und erhöhte Genauigkeit in ausgewählten Aspekten wie Datenerfassung, Analyse und Visualisierung ermöglichen. Dabei kann die Rolle dieser Technologien als ergänzendes Instrument betrachtet werden, welches lokale Gegebenheiten berücksichtigt, um eine ausgewogene und sinnvolle Anwendung in der Praxis zu gewährleisten.

### 1.4 Methodik und Vorgehensweise

Um die Forschungsfrage beantworten zu können, wird eine Vorgehensweise verfolgt, die verschiedene wissenschaftliche Methoden kombiniert. Die Methodik gliedert sich in vier Hauptphasen: Recherche und Analyse, Entwicklung vom Prototyp, die Anwendung und Diskussion.

In der ersten Phase, der Recherche, werden relevante Reglemente, Artikel und Berichte analysiert, um ein Verständnis der Grundlagen und Prozesse von Machbarkeitsstudien sowie der Rolle digitaler Tools und Technologien zu erlangen. Dabei werden sowohl theoretische Konzepte als auch praktische Anwendungen und Beispiele berücksichtigt. Vor allem wurde hier überlegt, mit welcher Programmiersprache das Ziel erreicht werden kann. Die Analyse der Technologien auf dem Markt fokussiert sich auf die Identifikation und Untersuchung von bestehenden Tools. Durch die Erfassung dieser Werkzeuge sollen ihre Anwendungsbereiche, Vor- und Nachteile dargestellt werden.

In der zweiten Phase der Arbeit liegt der Fokus auf der Entwicklung der digital unterstützten Machbarkeitsstudie. Hier wird mit dem Versuch begonnen, die Machbarkeitsstudie mit Hilfe digitaler Möglichkeiten so weit es Sinn macht zu automatisieren. Exemplarisch werden eigens erarbeitete Skripte eingesetzt, um Abschnitte der Machbarkeitsstudie zu unterstützen und optimieren. Ziel ist es, die Effizienz und Genauigkeit der Studie durch den Einsatz digitaler Technologien zu verbessern.

In der dritten Phase wird ein Workflow aufgezeigt, unter Einbezug der Skripts, sowie den Erkenntnissen aus Kapitel 4. Hierbei werden sowohl erfolgreiche Beispiele als auch Herausforderungen und Grenzen dargestellt.

In der letzten Phase werden Ergebnisse und Erkenntnisse zusammengefasst, die Forschungsfrage diskutiert und Empfehlungen für die Zukunft gegeben. Die wichtigsten Erkenntnisse aus den verschiedenen Phasen werden reflektiert und zentrale Aspekte noch weiter unterstrichen.

## 2 Grundlagen einer Machbarkeitsstudie

### 2.1 Definition und Zweck

Eine Machbarkeitsstudie ist eine systematische Untersuchung der Realisierbarkeit eines Bauprojektes, die dazu dient, mögliche Risiken, Chancen und Herausforderungen schon früh identifizieren und bewerten zu können. In der Schweiz spielt diese Studie eine entscheidende Rolle, da die Projekte den Anforderungen der Gemeinden gerecht werden, aber auch rentabel sein müssen.

Die Studie umfasst die Untersuchung verschiedener Faktoren eines Bauprojekts, wie die technische Machbarkeit, Umweltauswirkungen, wirtschaftliche Rentabilität, baurechtliche Anforderungen sowie soziale und politische Eingliederung. Eine umfassende Machbarkeitsstudie beinhaltet auch die Analyse des Marktpotenzials, die Standortanalyse und die Identifizierung von Risiken. Zusätzlich wird in der Regel eine detaillierte Finanzanalyse durchgeführt, um die erforderlichen Investitionen, die erwarteten Erträge und die möglichen Finanzierungsquellen für das Projekt schon früh ermitteln zu können. Die Durchführung einer Machbarkeitsstudie in der Schweiz erfordert häufig die Zusammenarbeit verschiedener Fachleute wie Architekten, Bauingenieure und Rechtsanwälte, um eine umfassende und fundierte Bewertung des Projektes zu gewährleisten. Zudem ist es wichtig, die relevanten Stakeholder in den Prozess einzubeziehen, um deren Bedenken und Anforderungen zu berücksichtigen und so die breite Akzeptanz für das Projekt zu fördern. Die frühzeitige Identifikation und Bewertung möglicher Herausforderungen einer Parzelle ermöglicht fundierte Entscheidungen und trägt dazu bei die Erfolgchancen eines Bauprojekts zu erhöhen. Eine gründliche Machbarkeitsstudie hilft also dabei, wertvolle Ressourcen wie Zeit und Geld zu sparen. Hierzu muss jedoch auch erwähnt werden, dass nicht alle Herausforderungen schon zu Beginn gelöst oder gefunden werden können.

### 2.2 Prozess einer Machbarkeitsstudie

In diesem Kapitell werden die Schritte kurz erläutert, die typischerweise nacheinander in der Machbarkeitsstudie Erhebung erfolgen:

1. **Projektdefinition** (1-2 Wochen): Zuerst muss das Projekt definiert werden. Um welche Parzelle handelt es sich? Welches sind die zu erreichenden Ziele? Was sind zu Beginn die Einschränkungen?
2. **Standortanalyse** (2-3 Wochen): Die relevanten Daten zur Parzelle werden erfasst. Wie liegt das Grundstück? Wie ist die Umgebung beschaffen? Wie sind die Zugänge und Infrastrukturen? Wie sehen die Grenzen aus?
3. **Analyse der gesetzlichen Rahmenbedingungen** (2-3 Wochen): Die relevanten gesetzlichen Unterlagen wie Baugesetze, Ortsbildschutz, Naturschutz usw. werden eingesehen.
4. **Technische Machbarkeit** (2-3 Wochen): In diesem Schritt wird geprüft, ob das Projekt technisch umsetzbar ist. Dazu gehört die genaue Prüfung der Bauvorschriften, damit erörtert werden kann, wie viel gebaut werden kann und wo Einschränkungen im Zusammenhang mit dem Baureglement liegen.

5. **Wirtschaftliche Machbarkeit (2-3 Wochen):** Hier wird analysiert, ob das Projekt wirtschaftlich umsetzbar ist, in Bezug auf die technische Machbarkeit.
6. **Gesellschaftliche Machbarkeit (2-3 Wochen):** Hier wird analysiert, ob das Projekt gesellschaftlich akzeptabel ist. Das beinhaltet, die Auswirkungen auf die Gemeinschaft und evtl. Umfragen. Dieser Prozess ist ausgehend von der technischen und wirtschaftlichen Machbarkeit.
7. **Entwurf von Konzeptvarianten (3-4 Wochen):** Basierend auf den vorherigen Schritten werden verschiedene Konzepte erstellt. Dieser Schritt kann auch nur 1 Konzept enthalten. Was dazu führt, dass er weniger Zeit in Anspruch nimmt.
8. **Bewertung der Konzeptvarianten (1-2 Wochen):** Anschliessend wird das Konzept anhand der Kriterien aus den vorherigen Schritten bewertet.
9. **Berichterstattung (2-3 Wochen):** Schliesslich wird ein Bericht über die Machbarkeitsstudie erstellt, der alle gesammelten Daten enthält und aufzeigt. Dieser Bericht kann dann den Stakeholdern vorgelegt werden.

Es ist zu beachten, dass jede Machbarkeitsstudie unterschiedlich ist und auf den spezifischen Kontext des Projekts zugeschnitten sein muss. Diese Schritte sind daher allgemein gemeint und variieren je nach Projekt (Abb. 1). Auch die Zeitangaben sind Schätzungen und können länger oder kürzer sein, es hängt von der Anzahl der Personen, die beteiligt sind, deren Erfahrungen und Grösse des Projektes ab, wie lange eine gesamtheitliche Studie dauert. Der generelle Konsens liegt daher zwischen 3 und 6 Monate in der Schweiz. Auch ist anzumerken, dass die gesamtheitliche Analyse von der technischen Machbarkeit abhängig ist und es somit Sinn macht dies zu automatisieren.



Abb. 1: Grafik Prozess

### 2.3 Zuvor erhobene Daten

Die Abwicklung einer Machbarkeitsstudie in der Schweiz erfordert die Erhebung und Untersuchung vielerlei Daten und Dokumente. Dies ist wichtig, um schon früh Potenziale zu erkennen und faktisch korrekte Entscheidungen treffen zu können. Ein massgebender Aspekt ist vor allem das Verständnis der jeweiligen Baureglemente. Diese sind auf kommunaler Ebene geregelt. Sie enthalten Informationen über zulässige Nutzungen und Bauweise. Besonders bedeutsam für eine Machbarkeitsstudie sind Höhenbeschränkungen und Grenzabstände. Sie stellen schon früh sicher, dass das Erscheinungsbild einer Gemeinde erhalten bleibt und grenzen das maximale Bauvolumen einer Parzelle ein. Baureglemente können in der Regel auf der Website der jeweiligen Gemeinde gefunden werden, im Bereich „Bau- und Planungsamt“.

Ein weiteres entscheidendes Dokument ist der Katasterplan. Der Plan ist im Grunde eine Karte, welche die Grenzen und Eigentumsverhältnisse von Grundstücken zeigt.

Er wird verwendet, um das gesetzliche Grundeigentum und die Grundstücksgrenzen zu definieren. Die in einem Katasterplan enthaltenen Informationen sind Parzellenummer, Grenzen, Gebäude und Strassennummer. Katasterpläne können über Geoinformationssysteme (GIS) von den jeweiligen Kantonen eingesehen und heruntergeladen werden. Diese Katasterpläne, sofern es sich um vektorisierte PDFs handelt, können anschliessend in CAD-Softwares eingelesen werden.

In Kombination mit Baureglement und Katasterplan ist ein 3D-Modell vom gewachsenen Terrain hilfreich, um die Höhe eines geplanten Gebäudes in Bezug auf das umliegende Gelände zu bestimmen. Sie liefern und visualisieren zugleich die Höhenverhältnisse einer Parzelle, was für die Einhaltung der Vorschriften unerlässlich ist. Diese 3D-Modelle können von der Website des Bundesamtes für Landestopografie (Swisstopo) heruntergeladen werden. Sie bieten eine Vielzahl von digitalen Höhenmodellen an, darunter swissALTI3D, ein hochauflösendes digitales Geländemodell der Schweiz. Andernfalls können auch Höhenpläne vom GIS heruntergeladen werden. Dabei besteht der Nachteil, dass das 3D Modell manuell mit einer CAD-Software erstellt werden muss. Hier kann angefügt werden, dass es grundsätzlich möglich ist, ohne ein 3D-Höhenmodell zu arbeiten. Allerdings führt dies zu einer reduzierten Genauigkeit der Machbarkeitsstudie, was wiederum die Zuverlässigkeit des Gesamtes beeinflusst.

## 2.4 Datenerhebung aus der Machbarkeitsstudie

Die Durchführung erfordert eine sorgfältige Sammlung und Analyse der zuvor erhobenen Daten, um Entscheidungen bezüglich des Potenzials der Parzelle treffen zu können. Um ein Bauvorhaben zu bewerten, braucht es zahlreiche Analysen. Es ist auch von Wichtigkeit, dass alle Daten, die erhoben werden, zusammenhängen. Folgenden Daten werden in der Arbeit von dem Prototyp generiert:

- Anzahl der Wohnungen
- Volumen ( $m^3$ ) / Fläche ( $m^2$ )
- Ausnutzung in %
  - Zu bauende Erholungsfläche
  - Zu bauende Garagenplätze
- Kosten/ $m^3$  Gebäudevolumen (GV) Norm SIA 416
- Kosten/ $m^2$  Geschossfläche (GF) Norm SIA 416

Die **Anzahl der Wohnungen** ist entscheidend, um die Dichte und das Ausmass des geplanten Projekts zu ermitteln. Sie hilft dabei, die Wohnkapazität und die Bewohnerdichte innerhalb vom Gebäude abzuschätzen und ermöglicht es, potenzielle soziale und ökologische Auswirkungen zu bewerten.

Das **Volumen** und die **Fläche** liefern Informationen über die Grösse der Entwicklung und tragen zur Beurteilung der Raumplanung bei. Dabei ist es von Bedeutung, diese zu visualisieren, um die Informationen greifbarer zu machen.

Die **Ausnutzung in Prozent** gibt Aufschluss über die gesamtheitliche Effizienz der Raumplanung und ermöglicht es, Infrastrukturanforderungen abzuschätzen.

Die Kosten pro Kubikmeter Gebäudevolumen (**GV**) und pro Quadratmeter Geschossfläche (**GF**) gemäss der Norm SIA 416 sind wichtige Faktoren bei der Beurteilung der Wirtschaftlichkeit des Bauvorhabens. Auch wenn sie noch nicht durch Details gefestigt sind, ist es wertvoll eine grobe Einschätzung zur Verfügung zu haben.

## 2.5 Zusammenwirkung der Daten

Die während der Machbarkeitsstudie analysierten Richtlinien und Daten bilden ein ineinander greifendes Gesamtpaket an Erkenntnissen. Das Baureglement, der Katasterplan und das Geländemodell sind eng miteinander verknüpft, sie ergänzen und vervollständigen sich gegenseitig. Die Höhenbeschränkungen und Grenzabstände, die im Baureglement festgelegt sind, beeinflussen die Gestaltungsmöglichkeiten und die Ausnützung der Parzelle. Die auf dem Katasterplan dargestellten Grundstücksgrenzen und Flächennutzungen geben Aufschluss über die räumlichen Gegebenheiten, während das 3D-Geländemodell die Höhenverhältnisse und die Einbindung des geplanten Gebäudes in die Topografie verdeutlicht. Das alles muss wiederum baureglementkonform sein. Durch die Kombination der Dokumente lassen sich schon zu Beginn Aussagen machen, was die Anzahl der Wohnungen, Volumen, Wohnfläche, Ausnützung und Kosten betrifft. Diese ganzheitliche Perspektive trägt entscheidend dazu bei, das Zusammenwirken der verschiedenen Faktoren zu erkennen und ein optimales Gleichgewicht zwischen den technischen, ökologischen und wirtschaftlichen Aspekten des Projekts zu erreichen.

## 2.6 Abgrenzung der Automation in einer Machbarkeitsstudie

Eine Automatisierung kann einen deutlichen Beitrag zur Verbesserung der Effizienz und Genauigkeit leisten. Dennoch ist es entscheidend, die Grenzen der Automatisierung in diesem Prozess zu erkennen und zu verstehen. Machbarkeitsstudien sind multidimensionale Untersuchungen, die nicht nur technische und finanzielle Aspekte, sondern auch gesellschaftliche und örtliche Gegebenheiten berücksichtigen sollten. Diese Aspekte sind von wichtiger Bedeutung, da sie den tatsächlichen Wert und die Relevanz eines Bauprojekts für die betroffenen Gemeinden und die umliegende Umwelt in einen gefestigten Kontext setzen.

Im Kontext dessen sind Ansätze wie der Diskursethik-Ansatz von Jürgen Habermas relevant. Habermas betont, dass Entscheidungen, die das Gemeinwohl betreffen, durch einen demokratischen und inklusiven Diskurs getroffen werden sollten, in dem alle betroffenen Parteien ihre Interessen und Bedenken einbringen können (Czogalla, 2010). Im Hinblick auf die Machbarkeitsstudie würde eine vollständige Automatisierung den Raum für solche Diskurse einschränken und die Berücksichtigung menschlicher Erfahrungen und Werte, die für den Erfolg eines Bauprojektes notwendig sind, verringern. Da hinsichtlich der Technologien erstaunliche Fortschritte gemacht wurden, sollten sie als Unterstützung betrachtet werden, um Fachleuten Daten zur Verfügung zu stellen. In diesem Sinne kann die Automatisierung bestimmter Prozesse die Arbeit von Fachleuten ergänzen. Sie sollte aber nicht als Ersatz für menschliches Urteilsvermögen und Kreativität angesehen werden.

## 3 Effizienz in der Bauindustrie

### 3.1 Definition von Effizienz

Die Effizienz in der Bauindustrie bezieht sich auf den bestmöglichen Einsatz von Ressourcen wie Zeit, Arbeit, Material und Finanzen, um das Projekt erfolgreich umsetzen zu können. Die Reduzierung von Verschwendungen, die Verhinderung von Bauverzögerungen und die verbesserte Koordination zwischen den Beteiligten spielen eine wesentliche Rolle. Ein effizientes Bauprojekt zeichnet sich daher durch höhere Produktivität, geringere Kosten, verbesserte Nachhaltigkeit und höhere Zufriedenheit der Stakeholder aus. Der britische Architekt, Designer und Autor William McDonough betont in seinem Werk «Cradle to Cradle: Remaking the way we make things» die Notwendigkeit der Effizienz und Nachhaltigkeit in der Bauindustrie durch innovative und zirkuläre Ansätze zu fordern (McDonough & Braungart, 2002). Die Implementation digitaler Technologien, wie sie in dieser Arbeit diskutiert werden, können dazu beitragen, dieses Ziel zu erreichen. Dies, indem sie präzisere Prozesse in der Datenerfassung, Analyse und Visualisierung innovativ unterstützen.

### 3.2 Digitale Effizienz in vier Bereiche

In der Arbeit wird die These aufgestellt, dass eine Automation mit digitalen Technologien die Machbarkeitsstudie unterstützen kann. Um dies diskutieren zu können, muss zuerst erläutert werden, was Effizienz im Bauwesen ist und wie die Wertschöpfung definiert werden kann. Die vier Bereiche, die abgedeckt werden, sind digitale Daten, Automation, Netzwerke und digitaler Zugang (Abb. 2). Die elektronische Erhebung und Auswertung von Daten ermöglicht eine präzisere und effizientere Analyse von Geodaten und Vorschriften. Eine Automation ermöglicht den Einsatz von autonom arbeitenden Systemen, die den Planungsprozess beschleunigen und Fehler reduzieren. Auf die Machbarkeitsstudie bezogen, wäre das der Zusammenhang zwischen den erhobenen Daten und der Erstellung von 3D-Modelle. Netzwerke spielen eine entscheidende Rolle bei der Vernetzung und Synchronisation bisher getrennter Aktivitäten, wodurch eine effektivere Zusammenarbeit und Entscheidungsfindung erreicht wird. Schliesslich erlaubt der mobile Zugriff auf das Internet und interne Netze eine schnellere und flexiblere Kommunikation zwischen den verschiedenen Akteuren, was wiederum eine bessere Anpassungsfähigkeit und Reaktionsfähigkeit im Planungsprozess ermöglicht. Diese vier Bereiche sind eng miteinander verknüpft und können gemeinsam zur Schaffung eines effizienteren Machbarkeitsstudienprozesses beitragen und eine sinnvolle Anwendung in der Praxis garantieren.

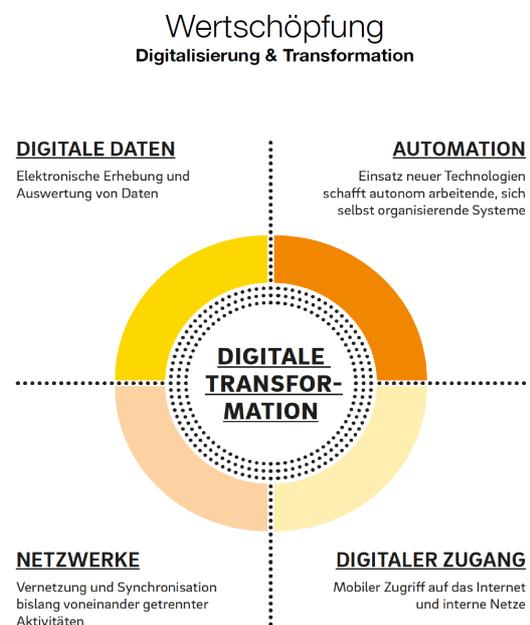


Abb. 2: Digitale Transformation

## 4 Technologien auf dem Markt

### 4.1 Bestehende Tools

Durch die Analyse der folgenden Werkzeuge sollen ihre Anwendungsbereiche, Vorteile und Nachteile erläutert werden. Mit dem Zweck den Nutzen von digitalen Mitteln in der Machbarkeitsstudie zu unterstreichen, aber auch um allenfalls negative Aspekte zu identifizieren.

#### 4.1.1 Sidewalk Labs

Sidewalk Labs ist ein Unternehmen, das sich mit der Entwicklung von Technologien und Lösungen für Städte beschäftigt. Es handelt sich um eine Tochtergesellschaft von Alphabet, die sich der Verbesserung der Lebensqualität in Städten durch die Anwendung von Technologie und Daten widmet. Sie arbeiten an Projekten, welche die Herausforderungen moderner Städte angehen, wie zum Beispiel die Erhöhung der Energieeffizienz und die Schaffung von nachhaltigen und bezahlbaren Wohnmöglichkeiten. Sidewalk Labs nutzt eine Vielzahl von Daten, um Muster zu erkennen und Probleme in Städten zu lösen. Dazu gehören Sensordaten von Strassen, Gebäuden und öffentlichen Verkehrsmitteln, sowie Daten aus sozialen Medien und anderen öffentlich zugänglichen Quellen. Sie setzen Analysemethoden ein, um aus diesen Daten Erkenntnisse zu gewinnen und Lösungen zu entwickeln, die den Bedürfnissen der jeweiligen Stadt entsprechen. Ein Produkt dieser Bemühungen ist Delve. Das ist ein Tool, welches maschinelles Lernen einsetzt, um das Potenzial auf grösseren Parzellen zu optimieren und zu bewerten (Abb. 3).

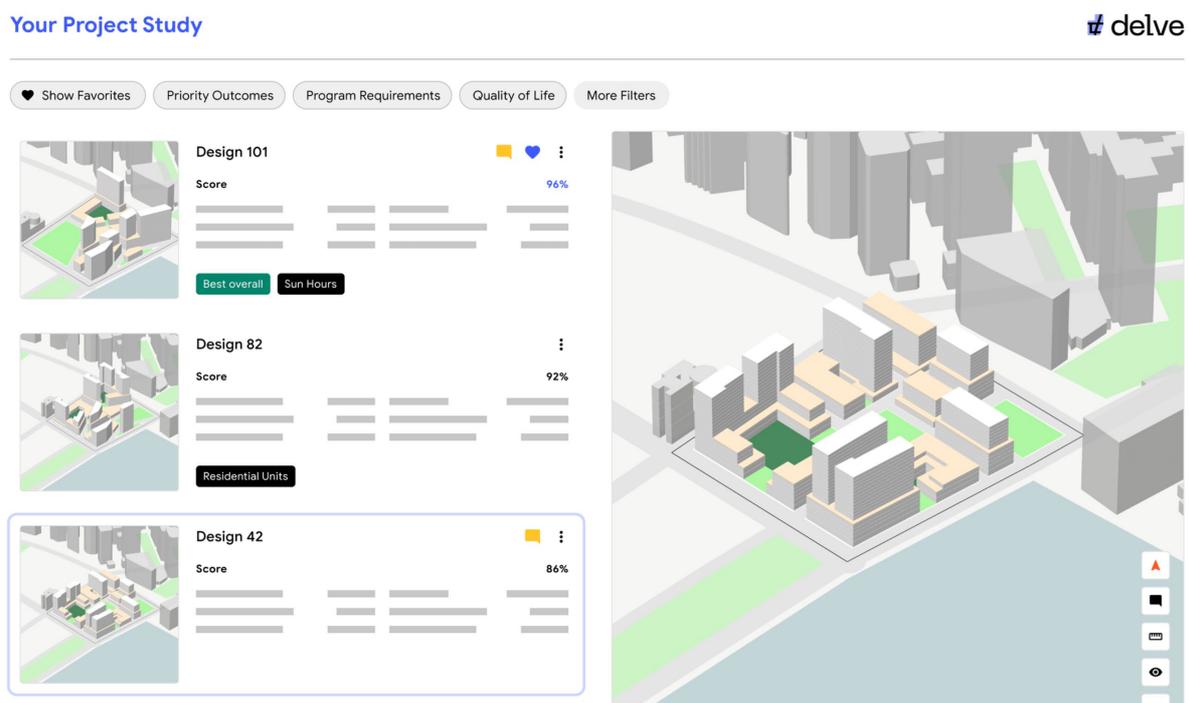


Abb. 3: Interface Delve

Nach einem Artikel im Internet hat Sidewalk Labs ihr geplantes Projekt "Quayside" in Toronto, Kanada, nicht mehr weiterverfolgt (Bozikovic, 2022). Das Projekt sah die Entwicklung eines innovativen, technologie- und datengetriebenes Wohn- und Arbeitsviertels vor. Die Entscheidung, das Projekt einzustellen, wurde aufgrund von Unstimmigkeiten zwischen dem Unternehmen und den Behörden in Bezug auf die Verarbeitung von Daten und die Regulierung des Projektes innerhalb von Sidewalk Labs getroffen. Es gab Bedenken bezüglich der Datensicherheit und Privatsphäre, insbesondere im Hinblick darauf, wer Zugang zu den gesammelten Daten haben würde und wie sie verwendet werden. Es gab auch Meinungsverschiedenheiten darüber, wie viel Entscheidungsgewalt das Unternehmen über die Entwicklung des Viertels haben sollte und ob es genug Regulierung geben würde, um sicherzustellen, dass die Interessen der Bewohner und der Stadt gestützt werden. Diese Entscheidung führte zu einem Rückschlag für Sidewalk Labs und die Vision einer "Smart City".

#### 4.1.2 Spacemaker

Spacemaker ist ein cloud-basiertes Planungswerkzeug, das von Autodesk entwickelt wurde (Abb. 4). Es soll Stadtplanern, Architekten und Immobilienentwicklern dabei helfen, datengesteuerte Entwürfe für Städte und grosse Überbauungen zu erstellen. Wie bei den anderen Tools, die hier beschrieben werden, ermöglicht die Kombination von künstlicher Intelligenz und maschinellem Lernen die effiziente Analyse und Optimierung von Projekten in Echtzeit. Es generiert verschiedene Entwurfsoptionen und deren Leistung hinsichtlich wichtiger Faktoren wie Sonneneinstrahlung, Lärm, Grünflächen, Wind und Sichtverbindungen. Ein wichtiger Vorteil von Spacemaker ist, dass der Planungsprozess beschleunigt wird, sowie gleichzeitig Entscheidungen erleichtert, indem das Ergebnis auf detaillierten Analysen beruht. Dies verbessert die Stadtplanung erheblich durch weniger Fehler und dem Einbinden von verschiedenen Interessensgruppen. Es bietet eine Diskussionsplattform für die Bewertung von Entwürfen. Allerdings ist Spacemaker, wie die anderen drei Tools, auch von der Qualität und Genauigkeit der bereitgestellten Daten abhängig. Unvollständige oder fehlerhafte Daten können zu ungenauen oder ineffizienten Entwürfen führen und auch zu gravierenden Fehlern, falls sie nicht identifiziert werden.

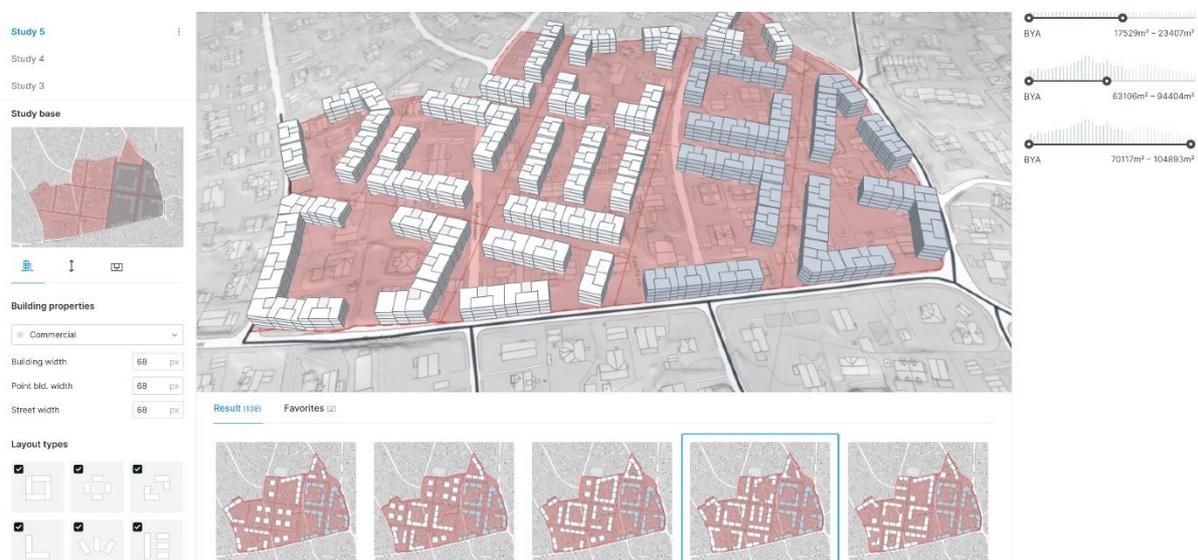


Abb. 4: Interface Spacemaker

### 4.1.3 Archistar

Archistar ist eine Software zur Unterstützung von Architekten und Immobilienentwicklern bei der Planung und Visualisierung der Machbarkeit (Abb. 5). Die Anwendung ist webbasiert und ermöglicht dem Benutzer, effizient eine Vielzahl von 3D Gebäudemodellen in Städten zu erstellen. Die Software benutzt dabei eine Kombination aus Machine-Learning und künstlicher Intelligenz, um schnell auf die relevanten Informationen zugreifen zu können. Eine Verknüpfung aus schon vorhanden Daten und generierten 3D-Modellen ermöglicht es Parzellen und Projekte fundierter bewerten zu können. Auch können nach Bedarf weitere Dokumente und Pläne importiert werden. Die Integration von Daten wie Geodaten und Bauvorschriften stellt sicher, dass die generierten Entwürfe den geltenden Normen und Vorschriften gerecht werden. Archistar kann dementsprechend für Städte interessant sein, weil es vor allem grössere

Parzellen generativ berechnen und in weniger Zeit viele potenzielle Entwürfe erstellen kann. Mit Archistar können Benutzer eine Vielzahl von Faktoren berücksichtigen, wie zum Beispiel topografische Bedingungen, Sonnenstand, Schattenwurf, Blickrichtungen, Windrichtungen und viele mehr. Genau das ist auch eine Schwachstelle, weil das Tool von der Stimmigkeit der



Abb. 5: Interface Archistar

vorhanden Daten anhängig ist und die Berechnungen auf diesen Daten basieren. In einer Stadt kann das eine Unmenge von Dokumenten sein, die verwaltet werden müssen. Für kleinere bis mittelgrosse Bauvorhaben wäre das Tool in der Schweiz aufgrund von Kosten und den Nutzen daraus nicht anwendbar. Auch machen es die vielen verschiedenen Vorgaben der zahlreichen Gemeinden deutlich schwieriger, wenn nicht fast unmöglich, Archistar ohne jeweilige Anpassung pro Gemeinde zu nutzen. Hinzu muss noch angefügt werden, dass Archistar im Moment primär von Australiern für Australien entwickelt wird.

### 4.2 Erkenntnisse aus den Tools

Alle recherchierten Tools beziehen sich hauptsächlich auf die Stadtplanung und deren Entwicklung. Wie aus dem „Quay Side“ Beitrag, schon hervorgeht, ist die Vision einer Smart City noch in weiter Ferne. Viele Normen und Gesetze stehen dem noch im Wege. Die recherchierten Tools sind zwar gut und dienen einem Ansatz, doch detailliert auf Schweizer Parzellen anwendbar sind sie nicht. Die Tools können mehrere Gebäude und Aufteilungen auf einer Parzelle innerhalb einer Stadt berechnen und erstellen, doch präzise Zahlen liefern für ein Mehrfamilienhaus in der Schweiz eher weniger. Somit liegt das Problem vor allem darin, dass sie für die Schweiz zu gross skaliert sind. Auch ermöglichen die Tools viele verschiedene Datenverarbeitungen, was es erschwert sie in der Schweiz für umsetzbare Projekte zu nutzen, da Sensordaten etc. allenfalls fehlen.

## 5 Entwicklung der digital unterstützten Machbarkeitsstudie

### 5.1 Auswahl und Begründung der eingesetzten Technologien

Zu Beginn wurden diverse Werkzeuge in Betracht gezogen, unter anderem Dynamo von Autodesk. Doch schliesslich fiel der Entscheid für die Entwicklung des Prototyps auf die Verknüpfung von Python und Grasshopper. Die Wahl basiert sich auf einer Reihe von Faktoren, die im Folgenden beschrieben werden.

Python bietet eine umfangreiche Standardbibliothek an Funktionen und zahlreiche externe Modulen, die kostenlos installiert werden können. Externe Bibliotheken werden von einer Open-Source Community entwickelt und sind für viele erdenkliche Anwendungsbereiche gedacht. Die grosse Vielfalt an verfügbaren Funktionen ermöglicht es, komplexe Berechnungen durchzuführen, aber auch die Manipulation von Datenformaten wie .DXF und .XLSX, die für die Machbarkeitsstudie relevant sind. Darüber hinaus ist die Syntax von Python im Gegensatz zu C++ oder JavaScript einfacher aufgebaut, was das gesamthafte Verständnis von einem längeren Code erleichtert.

Grasshopper dient als grafischer Algorithmus-Editor in Rhino und bietet eine hohe Flexibilität bei der Erstellung von parametrischen und generativen 3D-Modellen. Darum ist es auch möglich, ein Skript zu schreiben, das mit unterschiedlichen Inputs, wie zum Beispiel verschiedenen Parzellenformen zurechtkommen. Die Möglichkeit, schnell und einfach Anpassungen an den Algorithmen vorzunehmen, ermöglicht eine effiziente Iteration und Optimierung von Machbarkeitsmodellen. Ein Nachteil ist jedoch, dass Grasshopper eine Rhino-Lizenz erfordert, was die Benutzung vom Prototyp auf anderen Rechnern einschränkt.

Die Einbettung von Python in Grasshopper durch ein Python-Node bietet weitere Vorteile. Dadurch können die Stärken beider Werkzeuge kombiniert und genutzt werden, um eine geeignete Lösung zu entwickeln. Allerdings ist die aktuelle Implementierung des Python-Nodes in Grasshopper auf Python vers. 2.7 beschränkt, was einige Kompatibilitätsprobleme mit neueren Python-Bibliotheken verursachen kann (McNeel, 2020). Das hat zur Folge, dass allenfalls Umwege angedacht werden müssen, um das Ziel zu erreichen.

### 5.2 Datenintegration und Aufbereitung

#### 5.2.1 Baureglement

Das Baureglement ist ein wichtiges Dokument, in dem Gebäudelänge, Vollgeschosszahl, Grenzabstände, Ausnutzungsziffer und maximale Gebäudehöhe definiert sind. Das Problem ist, dass jede Gemeinde in der Schweiz diese Angaben anders angibt und beschreibt. Als Beispiel hat die Gemeinde Freienbach eine Tabelle, welche nach Zonen die genannten Informationen gliedert. Gemeinden, die eher noch mit älteren Baureglementen bauen, haben diese Angaben häufig in einem Lauftext beschrieben nach der entsprechenden Zone. Hombrechtikon, unter anderen, hat Absätze gemäss der entsprechenden Zone, die Gebäudelänge etc. mit Bullet Points angeben. Die Schweiz hat 2136 Gemeinden (Bundesamt für Statistik, 2023), was die Anzahl der Unterschiede zwischen den Baureglemente massiv vermehrt.

Diese Unterschiede machen es anspruchsvoll, die Baureglemente automatisch zu durchsuchen. Bei der Durchführung einer Wortsuche in Adobe Acrobat wird lediglich das gesuchte Wort angezeigt, was bei spezifischen Angaben wie beispielsweise "max. Gebäudelänge 40 m" problematisch ist. Bei einer Suche vom Wort „Gebäudelänge“ würden nur die Textstellen angezeigt werden. Wenn das Wort in einer Tabelle zu finden ist, muss trotzdem nach der Gebäudelänge manuell gesucht werden. Dies macht es wiederum fast unmöglich, den Prozess mit einem Skript zu automatisieren, ohne dass die Angaben alle gleich formatiert werden. Zwei Möglichkeiten wurden in dieser Arbeit untersucht. Eine als Experiment und eine für die Implementierung in das Skript. Bei der experimentellen Variante wurde das Baureglement von Lachen als Vorlagen genommen. Es wurde ein Python Skript geschrieben, das es ermöglicht ein Wort zu suchen und die umliegenden Sätze und Wörter ausgeben. Somit liegen auch Angaben aus einer Tabelle und ganze Sätze vor, um den Zusammenhang des gesuchten Wortes im Baureglement nachvollziehbar zu machen.

Zu Beginn wird der Pfad vom PDF angegeben (Abb. 6). Damit dieser nicht kopiert werden muss, kann der gesamte Computer durchsucht werden. Anschliessend wird der Begriff eingegeben, welcher gesucht werden soll. Diese Suche kann beliebig häufig wiederholt werden, um an weitere Angaben zu kommen.

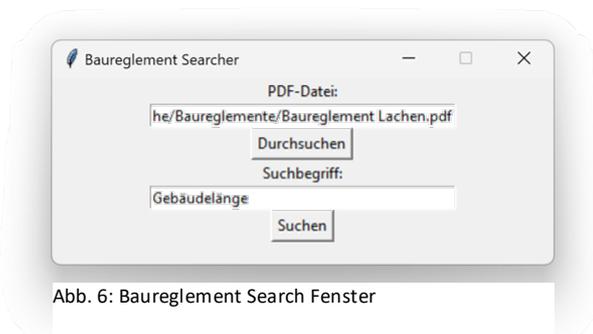


Abb. 6: Baureglement Search Fenster

Das Wort wird im PDF gesucht, dann wird ein Quadrat mit einer Seitenlänge von 7 cm um das Wort „gezeichnet“. Alles, was in dem Quadrat ist, wird ausgegeben (Abb. 7). Dies hat den Vorteil, dass das Reglement nicht vollständig gelesen werden muss. Jedoch

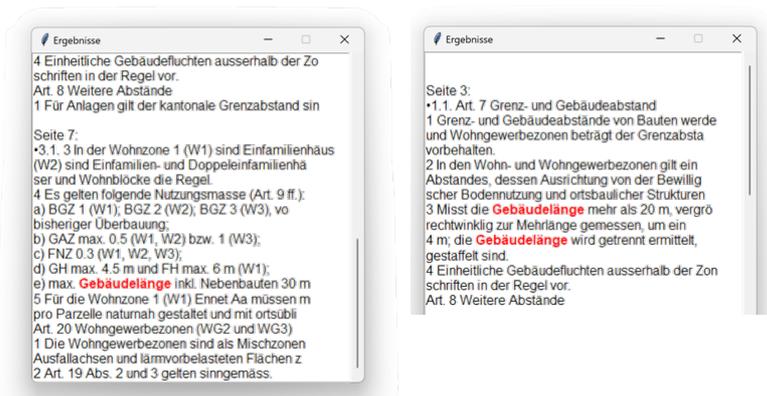


Abb. 7: Baureglement Search Ausgabe

liefert es noch zu wenig strukturierte Daten, um weiterarbeiten zu können. Um das zu lösen, wurden die jeweiligen Angaben (*Gemeinde inkl. Wohnzone, Kl. Grenzabstand, Gr. Grenzabstand, Gebäudehöhe, Vollgeschosszahl, Max. Gebäudelänge, Ausnutzungsziffer, Bemerkungen*) der Gemeinden in ein Excel geschrieben. Auch dieser Teil wurde mit dem Python Skript gelöst, damit ein Nutzer die Vorgaben in die Exceltabelle schreiben kann, ohne ein Excel aufmachen zu müssen.

Dem Benutzer wird ein Userinterface zur Verfügung gestellt, dass die Eingabe erlaubt (Abb. 8). Auf dieses Baureglement-Excel kann dann ein Python-Skript zugreifen, da die Angaben übersichtlich in einer Excel Tabelle strukturiert sind (Abb. 9). Auf diese Weise kann weitergearbeitet werden, um die Angaben, sofern es Sinn macht, automatisiert ins Grasshopper zu transferieren.



Abb. 8: Eingabefenster

Gemeinde inkl. Wohnzone	Kl. Grenzabstand	Gr. Grenzabstand	Gebäudehöhe	Vollgeschosszahl	Max. Gebäudelänge	Ausnutzungsziffer	Bemerkungen
Hombrechtikon KD	250,00	400,00	1400,00	2,00	3500,00	0,60	ABCD
Hombrechtikon KW	350,00	700,00	750,00	2,00	2500,00	-	ABCD
Hombrechtikon Z3/50	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon W1/20	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon W2/30	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon W2/35	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon W3/55	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon WG2/35	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon WG3/55	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Lachen K1	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen K2	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen KR	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen W1	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen W2	500,00	750,00	750,00	2,00	3000,00	0,30	Achtung! FNZ anstatt
Lachen W3	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen WS2	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen WG3	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Freienbach K	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Freienbach Z	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Freienbach WG4	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Freienbach WG3	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Freienbach W2	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Freienbach W3	600,00	1000,00	1000,00	3,00	4000,00	0,50	ABCD
Freienbach W4	250,00	400,00	1400,00	3,00	30,00	0,60	ABCD

Abb. 9: Baureglement Excel

### 5.2.2 Katasterplan

Der Katasterplan muss aufbereitet werden, damit er benutzt werden kann. Ein Katasterplan ist normalerweise ein vektorisiertes PDF. Im Gegensatz zu einer Bitmap-Grafik, die aus einer Ansammlung von Bildpunkten besteht, verwendet eine vektorbasierte Grafik mathematische Formeln, um Linien, Kurven und andere Formen zu definieren. Vektorbasierte Grafiken haben den Vorteil, dass sie skalierbar sind, ohne an Qualität zu verlieren. Das bedeutet, dass sie unabhängig von ihrer Größe immer scharf sein werden. Sie sind vektorbasiert, da sie so in CAD-Programme eingelesen werden können. Trotz der Vorteile eines vektorisierten PDFs ist es sinnvoll, das PDF in ein DXF-Dateiformat umzuwandeln. Der Hauptgrund dafür ist die Kompatibilität mit Python. Python bietet eine DXF-Manipulationsbibliothek „ezdxf“ an, welche die Arbeit mit dem Dateiformat ermöglicht. Da wie erwähnt das PDF auf Vektoren basiert ist es einfach, mit einem Gratis Online-Tool wie zum Beispiel zamzar.com die Datei zu konvertieren.

Für die Nutzung des Katasters in einer Automatisierung muss eine Logik entwickelt werden, welche es ermöglicht die gewünschte Parzelle zu erkennen. Das muss unabhängig von der Größe und Form der Parzelle immer funktionieren. Die Variante, die in diesem Skript ausgewählt wurde, ist die genaueste. Jedoch erfordert sie einen Mehraufwand. Es ist nötig, dass die Parzellengrenzlinien eine andere Farbe haben als der Rest des Katasterplans. So kann ausgewählt werden, welche Farbe behalten werden soll, um eine neue Datei zu erstellen nur mit der zu bearbeitenden Parzelle. Das kann mit einer DXF-Datei erreicht werden. Die Farbe der Linien können zuvor im PDF-Edit oder mit einem CAD in der DXF-Datei geändert werden.

### 5.2.3 Gewachsenes Terrain

Obwohl das Geländemodell bei der Erstellung einer Machbarkeitsstudie von grossem Nutzen ist, ist es nicht unbedingt das wichtigste Element. Es dient hauptsächlich dazu, die Machbarkeitsstudie genauer zu machen, indem die maximale Gebäudehöhe im frühen Stadium genau miteinander berechnet werden kann. Um das Geländemodell in einer Automatisierung wie Grasshopper weiterverarbeiten zu können ist es wichtig, dass es im richtigen Format vorliegt. Ein geeignetes Format ist das 3DM-Format, da es von Grasshopper unterstützt wird und es zahlreiche Werkzeuge und Plugins gibt, die es ermöglichen, das Geländemodell weiterzuverarbeiten. Das 3DM-Format bietet viele Vorteile gegenüber anderen Formaten, da es eine hohe Genauigkeit und Präzision bietet. Mit Grasshopper lässt sich das 3DM-Format automatisiert bearbeiten, um so Strukturen und Formen zu erstellen. Modelle können auf der Webseite von Swisstopo erhoben werden. Falls das Höhenmodell nicht dem gewünschten Detailgrad entspricht, ist es auch möglich, mit einem Geometer Plan und einem CAD-Programm einen höheren Detaillierungsgrad zu erlangen. Für die Lösung ist es unwichtig, woher das Modell kommt. Es muss lediglich eine 3DM-Datei sein und der Massstab vom Modell muss bekannt sein. Im besten Fall ist es der Gleiche, wie der Katasterplan. Im Prototyp ist es möglich, den Massstab im Nachhinein noch anzupassen, doch er muss bekannt sein, um ihn auf dieselbe Grösse wie der Katasterplan skalieren zu können.

### 5.3 Automatisierte Zusammenführung

Damit die Forschungsthese belegt werden kann, muss die Verknüpfung der verschiedenen Dokumente und Modelle automatisch geschehen. Die Erhebung sollte daher mit Sorgfalt durchgeführt werden, um die erhobenen Daten anschliessend für einen Mehrwert zu verwenden.

Die Machbarkeitsstudie verwendet verschiedene Datentypen wie Katasterpläne im DXF-Format, Bebauungsvorschriften in Excel-Tabellen und 3D-Modelle. Durch die Entwicklung von Algorithmen und deren Implementierung in Python und Grasshopper wird die Effizienz und Genauigkeit des Gesamtprozesses gesteigert. Der erste Schritt im automatisierten Integrationsprozess ist die Entwicklung von Algorithmen, welche die relevanten Daten effektiv filtern können. Der Python-Code in diesem Projekt wurde entwickelt, um Farbcodes aus DXF-Dateien zu extrahieren und Katasterpläne auf der Basis von Benutzereingaben zu filtern. Dieser Filterprozess ermöglicht die Isolierung bestimmter Datenelemente, wie die Grundstücksgrenzen, die für die Weiterführung wichtig sind. Dies, weil dann nur mit der Parzelle weitergearbeitet werden kann, die auch bearbeitet werden soll. Darüber hinaus werden im Projekt Algorithmen verwendet, um Daten aus Excel-Tabellen mit Bebauungsvorschriften zu lesen und zu verarbeiten. Durch die Nutzung der Datenmanipulationsfunktionen der Pandas-Bibliothek kann der Python-Code auf Basis von Benutzereingaben nach bestimmten Bebauungsvorschriften suchen und die entsprechenden Informationen ausgeben. Dieser automatisierte Prozess macht die manuelle Datenextraktion überflüssig, wodurch das Fehlerrisiko erheblich reduziert, und der Arbeitsablauf rationalisiert wird. Jedoch muss hier hinzugefügt werden, dass die Bauvorschriften nach wie vor manuell eingetragen werden müssen. Die automatische Analyse von Baureglementen ist anspruchsvoll, da alle 2136 Gemeinden in der Schweiz (Bundesamt für Statistik, 2023)

Unterschiede in der Struktur der Baureglements aufweisen. Als Beispiel verwalten die Gemeinden Hombrechtikon (Abb. 11) und Freienbach (Abb. 12) unterschiedliche Tabellen, wo sie die Masse angeben. Auch die verschiedenen Wohnzonen sind anders definiert. Es gibt auch Gemeinden, wie zum Beispiel Lachen (Abb. 10), welche die Grundmasse in einem Text anstatt einer Tabelle angeben. Das macht es schwierig, eine Logik für die automatisierte Analyse dieser Dokumente zu entwickeln. Hauptgrund ist, dass stets nach dem Gleichen gesucht werden muss, was jedoch meistens anders angegeben ist.

**Art. 7 Grenz- und Gebäudeabstand**

- 1 Grenz- und Gebäudeabstände von Bauten werden nach kantonalem Recht bestimmt. In den Wohn- und Wohngewerbebezonen beträgt der Grenzabstand mindestens 5 m. Art. 12 und 15 Abs. 3 bleiben vorbehalten.
- 2 In den Wohn- und Wohngewerbebezonen gilt ein grosser Grenzabstand von 150% des kantonalen Abstandes, dessen Ausrichtung von der Bewilligungsbehörde unter Berücksichtigung haushälterischer Bodennutzung und ortsbaulicher Strukturen bestimmt wird.
- 3 Misst die Gebäudelänge mehr als 20 m, vergrössert sich der Grenzabstand in den Wohnzonen, rechtwinklig zur Mehrlänge gemessen, um einen Viertel der Mehrlänge, höchstens jedoch um 4 m; die Gebäudelänge wird getrennt ermittelt, wenn Baukörper in der Tiefe um mindestens 2 m gestaffelt sind.
- 4 Einheitliche Gebäudefluchten ausserhalb der Zone W2 sind zwingend und gehen den Abstandsvorschriften in der Regel vor.

Abb. 10: Gemeinde Lachen Ausschnitt

**Grundmasse <sup>3)</sup>**

Zone	W1/20	W2/30	W2/35 WG2/35	W3/55 WG3/55
Vollgeschosszahl	max. 1	2	2	3
** Dachgeschoss	max. 2	2	2	2
Anrechenbare Untergeschosse	max. 1	1	1	1
Ausnutzungsziffer	max. 20 %	30 %	35 %	55 %
Überbauungsziffer	max. 30 %	30 %	30 %	30 %
- in den Wohnzonen			frei	frei
- in den WG-Zonen				
Kleiner Grundabstand	min. 4.0 m	4.0 m	4.0 m	6.0 m
Grosser Grundabstand	min. 9.0 m	9.0 m	9.0 m	12.0 m
Gebäudelänge / Gesamtlänge	max. 30.0 m	35.0 m	35.0 m	50.0 m
Gebäudehöhe	max. 4.8 m	7.5 m	7.5 m	10.5 m
Firsthöhe	max. 5.0 m	7.0 m	7.0 m	7.0 m

Die zu Wohn- und Arbeitszwecken genutzten Flächen in den Dach- und Untergeschossen dürfen zusammengezählt die zulässige Fläche eines durchschnittlichen Vollgeschosses nicht überschreiten.

Abb. 11: Gemeinde Hombrechtikon Ausschnitt

	K Art. 34	Z Art. 35	WG4 Art. 36	WG3 Art. 36	W4 Art. 37	W3 Art. 37	W2 Art. 37	L2* Art. 38	L2** Art. 38	G Art. 43	II Art. 44	II Art. 44	HZ Art. 46 Anhang 8
Anzahl Vollgeschosse	-	-	4	3	4	3	2	2	2	-	-	-	-
Ausnutzungsziffer (max. AZ)	-	-	0.65	0.60	0.65	0.60	0.45	0.30	0.30	-	-	-	0.70
wenn Gewerbeanteil mind. 25% ist oder wenn der Anteil preisgünstiger Wohnraum mind. 15% beträgt			+15%	+15%									
wenn der Anteil preisgünstiger Wohnraum mind. 15% beträgt					+10%	+10%	+10%						
										für Wohnen AZ 0.30, resp. mind. 1 Wohnung v von max. 150 m <sup>2</sup>			
Gebäudehöhe max.	10.50 m	13 m	13 m	10 m	13 m	10 m	7 m	7 m	7 m	13 m	20 m	13 m	10.50 m/ 13 m <sup>▲</sup>
Firsthöhe max.	15 m	17 m	17 m	14 m	17 m	14 m	10 m	10 m	10 m	17 m	20 m	17 m	15 m/ 17 m <sup>▲</sup>
Grenzabstand min.	50% der Gebäudehöhe, resp. mind. 3 m							10 m	5 m	50% der Gebäudehöhe, resp. mind. 3 m			
Kleiner Grenzabstand	-	-	60%							gegenüber W und WG			
Grosser Grenzabstand	-	-	100%		der Gebäudehöhe mind. 4 m								
Gebäudeabstand	-	-						20 m	10 m				
Gebäudelänge max.	-	-	40 m	40 m	40 m	40 m	40 m	20 m	30 m	60 m	-	-	-
Geschlossene Bauweise	-	gestattet	-	-	-	-	-	-	-	-	-	-	-
Wohnanteil	-	-	min. 30%	max. 70%	-	-	-	-	-	-	-	-	min. 30% max. 70%

<sup>▲</sup> gemäss Nutzungskonzept "Areal Steinfabrik" im Anhang 8

Abb. 12: Gemeinde Freienbach Ausschnitt

Die Integration dieser Algorithmen in Grasshopper, eine visuelle Programmiersprache für Rhino, ermöglicht eine nahtlose Verschmelzung von Datenanalyse und 3D-Modellierung. Durch die Integration von Python-Skripten in Grasshopper kann das Projekt die Generierung von 3D-Modellen auf der Grundlage der verarbeiteten Daten automatisieren. Dieser Ansatz spart nicht nur Zeit, sondern stellt auch sicher, dass die Modelle, die Bebauungsvorschriften und Katasterpläne genau wiedergeben. Zusammenfassend lässt sich sagen, dass durch die Automatisierung das Projekt einen effizienteren und genaueren Arbeitsablauf erreicht, der letztendlich zu einer qualitativ hochwertigeren Machbarkeitsstudie und dem architektonischen Entwurf führt.

## 5.4 Erstellung von 3D-Modellen

Die Erstellung von 3D-Modellen ist ein wichtiger Aspekt dieses Projekts, da sie eine effektive Visualisierung und Interpretation der baulichen Normen ermöglicht. Durch 3D-Darstellungen vom Möglichen können die Beteiligten die Auswirkungen und Durchführbarkeit von Projekten besser verstehen. Die Erstellung von 3D-Modellen in diesem Projekt beginnt mit den Daten, die durch den automatischen Integrationsprozess gewonnen werden. Wie in Abschnitt 5.3 erwähnt, verwendet das Projekt Algorithmen, um relevante Informationen aus verschiedenen Quellen wie DXF-Dateien und Excel-Tabellen zu extrahieren und zu verarbeiten. Diese aufbereiteten Daten dienen als Grundlage für die Erstellung von 3D-Modellen, welche die Bebauungsvorschriften und Katasterpläne wiedergeben.

Das Projekt verwendet Grasshopper, um die 3D-Modelle zu erstellen. Die parametrischen Fähigkeiten von Grasshopper ermöglichen die Erstellung von Geometrien auf der Grundlage von Eingabedaten vom Excelsheet wie Grenzabstände und Gebäudehöhen. Durch die Manipulation dieser Parameter kann der Anwender auf einfache Art und Weise die Auswirkungen von Entwurfsentscheidungen visualisieren. Neben der Erstellung der 3D-Modelle liegt ein weiterer Schwerpunkt des Projektes auf der Ausgabe von Informationen, die das Modell beschreiben. Dies wird im Grasshopper in Panels innerhalb eines Dashboards gemacht. Die Kombination von Informationen wie Ausnützung in %, Gebäudehöhe, Volumen vom Haus, etc. und der 3D-Darstellung soll die Anschaulichkeit und Verständlichkeit der Modelle verbessern können. Um die Einschränkungen und Möglichkeiten innerhalb des Untersuchungsgebietes klar darzustellen.

## 5.5 Skriptbeschreibung und Funktionsweise

### 5.5.1 Python: Baureglement Search Code

Der vorliegende Python-Code beinhaltet eine Anwendung zur Suche nach einem bestimmten Schlüsselwort innerhalb einer PDF-Datei, in diesem Fall die verschiedenen Baureglements der Gemeinden. Die Anwendung stellt eine grafische Benutzeroberfläche (GUI) mit Tkinter bereit, die es ermöglicht, eine PDF-Datei auszuwählen und ein Schlüsselwort einzugeben und nach diesem zu suchen. Um das Wort wird ein Quadrat mit der Seitenlänge 7 cm definiert. Alles, was sich innerhalb dieses Quadrates befindet, wird dann als Text ausgegeben, sowie das gesuchte Wort im Text rot hervorgehoben. Falls das Wort im PDF-Dokument mehrmals erscheint, werden mehrere Absätze ausgegeben.

```
1. #Import Bibliotheken
2. import pdfplumber
3. import tkinter as tk
4. from tkinter import filedialog, messagebox
5.
6. def search_pdf(file_path, keyword): 1. (Nummerierung löschen falls Code kopiert wird!)
7.     results = []
8.
9.     with pdfplumber.open(file_path) as pdf:
10.         for page in pdf.pages:
11.             words = page.extract_words(x_tolerance=3, y_tolerance=3)
12.             for word in words:
13.                 if word["text"].lower() == keyword.lower():
14.                     x0, y0, x1, y1 = word["x0"], word["top"], word["x1"], word["bottom"]
15.                     left, right = max(x0 - 105, 0), min(x1 + 105, page.width)
16.                     top, bottom = max(y0 - 105, 0), min(y1 + 105, page.height)
17.                     text = page.crop((left, top, right, bottom)).extract_text()
```

```

18.             if text:
19.                 results.append((text.strip(), page.page_number))
20.
21.     return results
22.
23. def display_results(results, keyword): 2. (Nummerierung löschen falls Code kopiert wird!)
24.     if not results:
25.         messagebox.showinfo("Ergebnisse", f"Keine Ergebnisse für \"{keyword}\" gefunden.")
26.         return
27.     root = tk.Tk()
28.     root.title("Ergebnisse")
29.     root.geometry("400x400")
30.     root.resizable(width=False, height=False)
31.
32.     scrollbar = tk.Scrollbar(root)
33.     scrollbar.pack(side=tk.RIGHT, fill=tk.Y)
34.
35.     text_box = tk.Text(root, font=("Helvetica", 12), yscrollcommand=scrollbar.set)
36.     text_box.pack(side=tk.LEFT, fill=tk.BOTH, expand=True)
37.
38.     scrollbar.config(command=text_box.yview)
39.
40.     for i, (text, page_num) in enumerate(results, start=1):
41.         text_box.insert(tk.END, f"\n\nSeite {page_num}:")
42.         paragraphs = text.split("\n\n") # Absätze aufteilen
43.         for j, paragraph in enumerate(paragraphs, start=1):
44.             text_box.insert(tk.END, f"\n{chr(8226)}{i}.{j}. {paragraph}")
45.             index = "1.0"
46.             while True:
47.                 index = text_box.search(keyword, index, tk.END)
48.                 if not index:
49.                     break
50.                 end_index = f"{index}+{len(keyword)}c"
51.                 text_box.tag_add("highlight", index, end_index)
52.                 index = end_index
53.             text_box.tag_config("highlight", foreground="red", font=("Helvetica", 12, "bold"))
54.
55.     tk.Button(root, text="OK", command=root.destroy).pack()
56.     root.mainloop()
57.
58. def browse_file(): 3. (Nummerierung löschen falls Code kopiert wird!)
59.     file_path = filedialog.askopenfilename(
60.         initialdir=".",
61.         title="PDF-Datei auswählen",
62.         filetypes=[("PDF", "*.pdf")]
63.     )
64.     if file_path:
65.         file_entry.delete(0, tk.END)
66.         file_entry.insert(0, file_path)
67.
68. def search_pdf_file(): 4. (Nummerierung löschen falls Code kopiert wird!)
69.     file_path = file_entry.get().strip()
70.     keyword = keyword_entry.get().strip()
71.     if not file_path or not keyword:
72.         messagebox.showerror("Fehler", "Bitte geben Sie eine PDF-Datei und einen Suchbegriff ein.")
73.         return
74.     try:
75.         results = search_pdf(file_path, keyword)
76.         display_results(results, keyword)
77.     except Exception as e:
78.         messagebox.showerror("Fehler", f"Fehler beim Durchsuchen der PDF-Datei:\n{e}")
79.
80. root = tk.Tk() 5. (Nummerierung löschen falls Code kopiert wird!)
81. root.title("Baureglement Searcher")
82. root.geometry("400x150")
83. root.resizable(width=False, height=False)
84.
85. file_label = tk.Label(root, text="PDF-Datei:")
86. file_label.pack()
87. file_entry = tk.Entry(root, width=40)
88. file_entry.pack()
89. file_button = tk.Button(root, text="Durchsuchen", command=browse_file)
90. file_button.pack()
91.
92. keyword_label = tk.Label(root, text="Suchbegriff:")
93. keyword_label.pack()
94. keyword_entry = tk.Entry(root, width=40)
95. keyword_entry.pack()
96.

```

```
97. search_button = tk.Button(root, text="Suchen", command=search_pdf_file)
98. search_button.pack()
99.
100. root.mainloop()
```

Der Code beginnt mit dem Import der notwendigen Bibliotheken:

- ``pdfplumber``: Zum Lesen der PDF-Dokumente
- ``tkinter``: Zur Erstellung vom GUI

Im Folgenden werden die Hauptfunktionen erläutert (Im Skript grün von oben nach unten):

1. ``search_pdf(file_path, keyword)``: Diese Funktion sucht in der angegebenen PDF-Datei (über den Dateipfad ``file_path``) nach dem Schlüsselwort (``keyword``). Sie gibt eine Liste von Ergebnissen zurück, die das gefundene Schlüsselwort sowie die jeweilige Seitennummer enthalten. Die Funktion öffnet zunächst die PDF-Datei und extrahiert die Wörter auf jeder Seite. Anschliessend wird jedes Wort auf Gleichheit mit dem gesuchten Schlüsselwort überprüft, wobei Gross- und Kleinschreibung ignoriert wird. Wenn das Schlüsselwort gefunden wird, wird ein Ausschnitt der Seite erstellt und der Text innerhalb dieses Ausschnitts extrahiert. Schliesslich wird der extrahierte Text zusammen mit der Seitennummer in die Ergebnisliste aufgenommen.
2. ``display_results(results, keyword)``: Diese Funktion zeigt die Ergebnisse der Suche in einem neuen GUI-Fenster an. Die gefundenen Textausschnitte und Seitennummern werden in einer Textbox angezeigt. Dabei wird das Schlüsselwort rot und fett hervorgehoben. Wenn keine Ergebnisse gefunden wurden, wird stattdessen eine Nachricht angezeigt, die darauf hinweist, dass keine Ergebnisse gefunden wurden.
3. ``browse_file()``: Diese Funktion öffnet einen Dateialog zum Auswählen einer PDF-Datei und aktualisiert das Eingabefeld für den Dateipfad in der GUI.
4. ``search_pdf_file()``: Diese Funktion ruft die ``search_pdf``-Funktion auf und übergibt den eingegebenen Dateipfad und das Schlüsselwort. Anschliessend werden die Ergebnisse an die ``display_results``-Funktion weitergegeben, um sie anzuzeigen. Wenn beim Durchsuchen der PDF-Datei ein Fehler auftritt, wird eine Fehlermeldung angezeigt.
5. Nach der Definition der Hauptfunktionen wird die GUI der Anwendung erstellt. Dazu wird ein ``tk.Tk()``-Objekt (genannt ``root``) initialisiert und konfiguriert. Die GUI besteht aus Eingabefeldern zum Auswählen der PDF-Datei und zum Eingeben des Schlüsselworts sowie Schaltflächen zum Durchsuchen der Datei und zum Starten der Suche. Das Skript endet mit dem Aufruf der ``root.mainloop()``-Methode, die die GUI startet und die Anwendung ausführt.

### 5.5.2 Python: Katasterplan-Analyse und Baureglement-Verarbeitungstool

Dieses Skript wird zur Analyse von Katasterplänen im DXF-Format und zur Verarbeitung von Baureglementdaten aus Excel-Dateien verwendet. Das Skript implementiert eine grafische Benutzeroberfläche (GUI) mit PySimpleGUI, die es ermöglicht eine DXF-Datei auszuwählen und nach Farbcodes zu filtern, welche verschiedene Grundstücksgrenzen repräsentieren. Anschliessend werden Baureglementinformationen aus einer Excel-Datei abgerufen und in einer neuen Datei gespeichert. Schliesslich bietet das Skript die Möglichkeit, ein zugehöriges

Grasshopper-File zu öffnen, das für die weitere Verarbeitung der Daten und 3D-Modelle genutzt werden kann.

```

1. #Alle importierte Bibliotheken
2. import ezdxf
3. import os
4. import pandas as pd
5. import PySimpleGUI as sg
6.
7. #Pfade zur Desktopumgebung & Folder 1. (Nummerierung löschen falls Code kopiert wird!)
8. desktop_path = os.path.join(os.path.expanduser("~"), "Desktop")
9. folder_name = "Machbarkeitsstudie Prototype Tool"
10. output_folder = os.path.join(desktop_path, folder_name)
11. excel_input_file = os.path.join(output_folder, "Excel Baureglementdaten.xlsx")
12. excel_output_file = os.path.join(output_folder, "Excel BaureglementdatenOutput.xlsx")
13. grasshopper_file = os.path.join(output_folder, "Prototype Katasterplan.gh")
14.
15. #PysimpleGui Erstellung 2. (Nummerierung löschen falls Code kopiert wird!)
16. sg.theme('DarkGrey6')
17. layout = [
18.     [sg.Text('Der gesamte Ordner, mit allen Dateien muss auf dem Desktop gespeichert sein!',
font=("Helvetica", 12, "bold"), text_color='tomato')],
19.     [sg.Text('Wählen Sie den Pfad zum ursprünglichen Kataster in DXF:')],
20.     [sg.Input(key='-PATH-', size=(90, 1), readonly=True, text_color='black'),
sg.FileBrowse('Durchsuchen', target='-PATH-', file_types=(('DXF Files', '*.dxf')))],
21.     [sg.Button('Weiter')]]
22. ]
23. window = sg.Window('Wählen Sie ein DXF-File', layout)
24.
25. #Schleife bis Fenster geschlossen wird
26. while True:
27.     event, values = window.read()
28.     if event == sg.WIN_CLOSED:
29.         break
30.     elif event == 'Weiter':
31.         selected_path = values['-PATH-']
32.         layout_dxf_path = selected_path
33.         break
34. window.close()
35.
36. #Farbcodes im DXF Kataster finden 3. (Nummerierung löschen falls Code kopiert wird!)
37. def extract_color_codes(file_path):
38.     dwg = ezdxf.readfile(file_path)
39.     msp = dwg.modelspace()
40.     color_codes = set()
41.     for entity in msp:
42.         if entity.dxftype() in ['LINE', 'ARC']:
43.             color_codes.add(entity.dxf.color)
44.         elif entity.dxftype() == 'LWPOLYLINE':
45.             for segment in entity:
46.                 if isinstance(segment, ezdxf.entities.DXFEntity):
47.                     color_codes.add(segment.dxf.color)
48.     return color_codes
49. if __name__ == '__main__':
50.     color_codes = extract_color_codes(layout_dxf_path)
51.     color_codes_list = [
52.         "1 = Rot",
53.         "2 = Gelb",
54.         "3 = Grün",
55.         "4 = Cyan",
56.         "5 = Blau",
57.         "6 = Magenta",
58.         "7 = Schwarz",
59.         "8 = Grau",
60.     ]
61. #Fenster für DXF Farbe Ausgabe 4. (Nummerierung löschen falls Code kopiert wird!)
62.     layout = [

```

```

63.         [sg.Text(f'Die verschiedenen Farben im DXF sind: {color_codes}', font=("Helvetica",
12, "bold"), text_color='white')],
64.         [sg.Listbox(color_codes_list, size=(20, 10), enable_events=True, key='-LIST-')],
65.         [sg.Button('Weiter')]
66.     ]
67.     window = sg.Window('Farbcodes im DXF', layout, finalize=True)
68.     while True:
69.         event, values = window.read()
70.         if event == sg.WIN_CLOSED or event == 'Weiter':
71.             break
72.     window.close()
73.
74. #DXF Filterung nach Farbcode 5. (Nummerierung löschen falls Code kopiert wird!)
75. #Pfad Input-Datei
76. input_file = layout_dxf_path
77. #Pfad Output-Datei (Parzelle)
78. output_file = os.path.join(output_folder, "Parzelle gefiltert.dxf")
79. #Farbcode für Linien (Input)
80. color_code = sg.popup_get_text("Bitte geben Sie den gewünschten Farbcode ein der
beibehalten wird: ")
81. color_code = int(color_code) if color_code.isdigit() and int(color_code) in color_codes
else None
82. if color_code is None:
83.     sg.popup("Der eingegebene Farbcode ist ungültig oder nicht vorhanden.")
84. doc = ezdxf.readfile(input_file)
85. #Hole die Model-Space-Entitäten
86. msp = doc.modelspace()
87. #Filtere alle Linien mit dem angegebenen Farbcode
88. filtered_entities = []
89. for entity in msp:
90.     if entity.dxf.type() in ["LINE", "ARC"] and entity.dxf.color == color_code:
91.         filtered_entities.append(entity)
92. #Überprüfen, ob die Eingabezahl im Set der Farbcodes enthalten ist
93. if color_code in color_codes:
94.     #Erstelle ein neues DXF-File mit den gefilterten Linien
95.     doc2 = ezdxf.new()
96.     msp2 = doc2.modelspace()
97.     for entity in filtered_entities:
98.         if entity.dxf.type() == "LINE":
99.             msp2.add_line((entity.dxf.start[0], entity.dxf.start[1]), (entity.dxf.end[0],
entity.dxf.end[1]),
100.                            dxfattribs={'color': entity.dxf.color})
101.         elif entity.dxf.type() == "ARC":
102.             msp2.add_arc(center=(entity.dxf.center[0], entity.dxf.center[1]),
radius=entity.dxf.radius,
103.                          start_angle=entity.dxf.start_angle,
end_angle=entity.dxf.end_angle,
104.                          dxfattribs={'color': entity.dxf.color})
105.     #Speichere das neue DXF-File
106.     doc2.saveas(output_file)
107.     sg.popup(f"Das DXF wurde gespeichert unter: {output_file}")
108. else:
109.     sg.popup("Die eingegebene Farbcode ist nicht im DXF vorhanden. Das DXF wird nicht
gespeichert.")
110.
111. #Excel Daten Ausgabe der Gemeinden 6. (Nummerierung löschen falls Code kopiert wird!)
112. #Lesen der Daten aus der Excel-Datei in ein pandas DataFrame
113. df = pd.read_excel(excel_input_file)
114. #Spaltennamen
115. df.columns = ['Gemeinde inkl. Wohnzone', 'Kl. Grenzabstand', 'Gr. Grenzabstand',
'Gebäudehöhe', 'Vollgeschosszahl', 'Max. Gebäudelänge', 'Ausnutzungsziffer', 'Bemerkungen']
116. #Eingabe
117. eingabe = sg.popup_get_text("Bitte geben Sie den Namen der Gemeinde inkl. die Wohnzone ein:
")
118. #Suche der Zeile nach der Eingabe
119. row = df.loc[df['Gemeinde inkl. Wohnzone'] == eingabe]
120. if not row.empty:
121.     kl_grenzabstand = row.iloc[0]['Kl. Grenzabstand']
122.     gr_grenzabstand = row.iloc[0]['Gr. Grenzabstand']
123.     gebauehoehe = row.iloc[0]['Gebäudehöhe']

```

```

124.     vollgeschosszahl = row.iloc[0]['Vollgeschosszahl']
125.     max_gebaudelaenge = row.iloc[0]['Max. Gebäudelänge']
126.     ausnutzungsziffer = row.iloc[0]['Ausnutzungsziffer']
127.     bemerkungen = row.iloc[0]['Bemerkungen']
128. #Ausgabe der Gemeindedaten
129.     output_text = f""Gemeinde inkl. Wohnzone: {eingabe}
130. Kl. Grenzabstand: {kl_grenzabstand}
131. Gr. Grenzabstand: {gr_grenzabstand}
132. Gebäudehöhe: {gebaeudehoehe}
133. Vollgeschosszahl: {vollgeschosszahl}
134. Max. Gebäudelänge: {max_gebaudelaenge}
135. Ausnutzungsziffer: {ausnutzungsziffer}
136. Bemerkungen: {bemerkungen}
137. ""
138.     sg.popup_scrolled(output_text)
139. else:
140.     sg.popup("Die eingegebene Gemeinde ist nicht in der Liste enthalten.")
141. #Erstellen eines neuen DataFrames 7. (Nummerierung löschen falls Code kopiert wird!)
142. output_df = pd.DataFrame({
143.     'Gemeinde inkl. Wohnzone': [eingabe],
144.     'Kl. Grenzabstand': [kl_grenzabstand],
145.     'Gr. Grenzabstand': [gr_grenzabstand],
146.     'Gebäudehöhe': [gebaeudehoehe],
147.     'Vollgeschosszahl': [vollgeschosszahl],
148.     'Max. Gebäudelänge': [max_gebaudelaenge],
149.     'Ausnutzungsziffer': [ausnutzungsziffer],
150.     'Bemerkungen': [bemerkungen]
151. })
152. #Exportieren des DataFrames in eine neue Excel-Datei
153. output_df.to_excel(excel_output_file, index=False)
154. sg.popup("Das Excel wurde neu gespeichert mit den gesuchten Baureglement Daten")
155.
156. #Grasshopper Öffnung 8. (Nummerierung löschen falls Code kopiert wird!)
157. def open_grasshopper_file(file_path):
158.     user_input = sg.popup_get_text("Möchten Sie das Grasshopper-File öffnen? (ja/nein):
159. ").lower()
160.     if user_input == "ja":
161.         sg.popup("Das Grasshopper-File wird geöffnet")
162.         os.startfile(file_path)
163.     elif user_input == "nein":
164.         sg.popup("Das Grasshopper-File wird nicht geöffnet")
165.     else:
166.         sg.popup("Ungültige Eingabe, bitte geben Sie 'ja' oder 'nein' ein.")
167. open_grasshopper_file(grasshopper_file)

```

Der Code beginnt mit dem Import der notwendigen Bibliotheken:

- `ezdxf` : Zum Lesen, Schreiben und Manipulieren von DXF-Dateien
- `os` : Bietet eine Möglichkeit, mit dem Betriebssystem zu interagieren
- `pandas` : Excel, Datenverarbeitungs- und Analysebibliothek für Python
- `PySimpleGUI` : Zur Erstellung von grafischen Benutzeroberflächen (GUIs) in Python

Das Skript enthält verschiedene Funktionen, die alle zu einer Anwendung zusammengeführt werden. Die Hauptteile werden im folgenden Abschnitt erläutert (Im Skript grün von oben bis unten):

**1. Festlegen von Pfaden:** Das Skript legt Pfade für Desktopumgebung, Dateien und Ordner fest. Es prüft, ob der Ordner "Machbarkeitsstudie Prototype Tool" auf dem Desktop vorhanden ist und speichert darin verschiedene Dateien wie Excel-Dateien für Baureglementdaten und Grasshopper-Dateien.

**2. Erstellen einer grafischen Benutzeroberfläche (GUI) mit PySimpleGUI:** Die GUI ermöglicht es dem Benutzer, den Pfad zum ursprünglichen Kataster in DXF-Dateiformat auszuwählen und mit den weiteren Schritten fortzufahren. Das Layout enthält Textelemente, Eingabefelder, Schaltflächen und Listenfelder, die zusammen eine intuitive Benutzeroberfläche bilden.

**3. Farbcodes im DXF-Kataster extrahieren:** Das Skript definiert eine Funktion namens `extract_color_codes`, die Farbcodes aus dem gewählten DXF-Kataster extrahiert und in einem Set speichert. Diese Funktion liest die DXF-Datei und durchläuft alle Entitäten (Linien, Bögen und Polylinien), um die jeweiligen Farbcodes zu extrahieren und zu speichern.

**4. Fenster für DXF-Farbausgabe:** Nachdem die Farbcodes extrahiert wurden, erstellt das Skript ein neues Fenster, das die verschiedenen Farbcodes im DXF anzeigt und dem Benutzer die Möglichkeit gibt, einen bestimmten Farbcode auszuwählen. Diese Auswahl wird später verwendet, um die DXF-Datei zu filtern.

**5. DXF-Filterung nach Farbcode:** Das Skript filtert die DXF-Datei basierend auf dem vom Benutzer eingegebenen Farbcode und erstellt eine neue DXF-Datei, die nur die gefilterten Linien, Bögen und Polylinien enthält. Diese Funktion geht durch alle Elemente gesamten DXF, sucht diejenigen heraus, die eine bestimmte Farbe haben, und speichert sie in einer Liste namens `filtered_entities`. Anschliessend wird eine neue DXF-Datei erstellt, die die gefilterten Entitäten enthält, und sie wird im zuvor festgelegten Ausgabeordner auf dem Desktop gespeichert.

**6. Excel-Daten Ausgabe der Gemeinden:** Das Skript liest die Daten aus einer Excel-Datei, die Baureglementdaten von verschiedenen Gemeinden enthält und speichert sie in einem Pandas Data Frame. Der Benutzer wird aufgefordert den Namen der gewünschten Gemeinde und Wohnzone einzugeben und das Skript sucht nach den entsprechenden Daten in der Tabelle. Wenn die eingegebene Gemeinde gefunden wird, werden alle relevanten Daten, die in der Reihe sind, angezeigt.

**7. Erstellen eines neuen Data Frames und Exportieren in eine neue Excel-Datei:** Nachdem die gewünschten Daten gefunden wurden, erstellt das Skript ein neues Pandas Data Frame, das nur die ausgewählten Daten enthält, und exportiert es in eine neue Excel-Datei, die im Ausgabeordner gespeichert wird.

**8. Grasshopper-Datei öffnen:** Schliesslich bietet das Skript dem Benutzer die Möglichkeit eine Grasshopper-Datei zu öffnen, die für die Bearbeitung vom 3D-Modell verwendet werden kann. Wenn der Benutzer zustimmt, wird die Grasshopper-Datei geöffnet, andernfalls wird sie nicht geöffnet.

### 5.5.3 Python: Eingabe Tool Baureglementdaten

Dieses Skript bietet ein einfaches Tool zur Eingabe von den Baureglementdaten, die anschliessend mit dem zuvor erklärtem Skript wieder abgerufen werden können. Das Skript bietet eine grafische Benutzeroberfläche zum Eingeben von Baureglementdaten, wie Gemeinde- und Wohnzonennamen, kleiner und grosser Grenzabstand, Gebäudehöhe, Vollgeschosszahl, maximale Gebäudelänge und Bemerkungen.

Nach dem Speichern der eingegebenen Daten fügt das Skript sie in eine bestehende Excel-Datei ein, die im vorgegebenen Ausgabeordner gespeichert ist.

```

1. import os
2. import PySimpleGUI as sg
3. import openpyxl
4.
5. #Pfad zum Desktop des Benutzers (Variabel) 1. (Nummerierung löschen falls Code kopiert wird!)
6. desktop_path = os.path.join(os.path.expanduser("~"), "Desktop")
7.
8. # Vorhandenen Ordner auf dem Desktop verwenden
9. folder_name = "Machbarkeitsstudie Prototype Tool"
10. output_folder = os.path.join(desktop_path, folder_name)
11.
12. # Setze den Pfad für die Excel-Datei mit den Baureglementdaten
13. excel_input_file = os.path.join(output_folder, "Excel Baureglementdaten.xlsx")
14.
15. #Layout des PySimpleGUI-Fensters 2. (Nummerierung löschen falls Code kopiert wird!)
16. layout = [
17.     [sg.Text("Gemeinde inkl. Wohnzone"), sg.InputText(key="gemeinde")],
18.     [sg.Text("Kl. Grenzabstand"), sg.InputText(key="kl_grenzabstand")],
19.     [sg.Text("Gr. Grenzabstand"), sg.InputText(key="gr_grenzabstand")],
20.     [sg.Text("Gebäudehöhe"), sg.InputText(key="gebaeudehoehe")],
21.     [sg.Text("Vollgeschosszahl"), sg.InputText(key="vollgeschosszahl")],
22.     [sg.Text("Max. Gebäudelänge"), sg.InputText(key="max_gebaedelaenge")],
23.     [sg.Text("Ausnutzungsziffer"), sg.InputText(key="ausnutzungsziffer")],
24.     [sg.Text("Bemerkungen"), sg.InputText(key="bemerkungen")],
25.     [sg.Button("Speichern"), sg.Button("Abbrechen")]
26. ]
27.
28. window = sg.Window("Eingabe Baureglementdaten", layout)
29.
30. while True:
31.     event, values = window.read() 3. (Nummerierung löschen falls Code kopiert wird!)
32.     if event == "Speichern":
33.         #Lade Excel-Datei und das Arbeitsblatt 4. (Nummerierung löschen falls Code kopiert wird!)
34.         workbook = openpyxl.load_workbook(excel_input_file)
35.         worksheet = workbook.active
36.
37.         # Finde die erste freie Zeile in der ersten Spalte
38.         first_empty_row = len(worksheet["A"]) + 1
39.
40.         # Füge die Werte in die entsprechenden Spalten der freien Zeile ein
41.         worksheet.cell(row=first_empty_row, column=1, value=values["gemeinde"])
42.         worksheet.cell(row=first_empty_row, column=2, value=values["kl_grenzabstand"])
43.         worksheet.cell(row=first_empty_row, column=3, value=values["gr_grenzabstand"])
44.         worksheet.cell(row=first_empty_row, column=4, value=values["gebaeudehoehe"])
45.         worksheet.cell(row=first_empty_row, column=5, value=values["vollgeschosszahl"])
46.         worksheet.cell(row=first_empty_row, column=6, value=values["max_gebaedelaenge"])
47.         worksheet.cell(row=first_empty_row, column=7, value=values["ausnutzungsziffer"])
48.         worksheet.cell(row=first_empty_row, column=8, value=values["bemerkungen"])
49.
50.         # Speichere die Änderungen in der Excel-Datei
51.         workbook.save(excel_input_file)
52.
53.         # Schliesse das Fenster und beende die Schleife
54.         sg.popup("Erfolg!", "Die Daten wurden erfolgreich in die Excel-Datei eingefügt.")
55.         break
56.     elif event in (sg.WIN_CLOSED, "Abbrechen"):
57.         break
58.
59. # Schliesse das Fenster
60. window.close()
61.

```

Der Code beginnt mit dem Import der notwendigen Bibliotheken:

- `os` : Bietet eine Möglichkeit, mit dem Betriebssystem zu interagieren

- `PySimpleGUI`: Zur Erstellung von grafischen Benutzeroberflächen (GUIs) in Python
- `openpyxl`: Zur Manipulation von Excel-Dateien innerhalb von Python

Die Hauptteile werden im folgenden Abschnitt erläutert (Im Skript grün von oben bis unten):

1. `os.path.join()` und `os.path.expanduser()`: Diese Funktionen werden verwendet, um Datei- und Verzeichnispfade auf plattformunabhängige Weise zu verbinden und den Pfad zum Home-Verzeichnis des aktuellen Benutzers zu ermitteln.
2. PySimpleGUI-Funktionen (`sg.Text()`, `sg.InputText()`, `sg.Button()`, `sg.Window()`): Diese Funktionen werden verwendet, um eine grafische Benutzeroberfläche (GUI) für die Erfassung von Baureglementdaten zu erstellen. Sie erstellen Textelemente, Eingabefelder, Schaltflächen und ein Fenster für die GUI. Die GUI ermöglicht es Benutzern, Daten einzugeben und sie entweder zu speichern oder die Aktion abzurechnen.
3. `window.read()`: Diese Methode wartet auf Benutzerinteraktionen und gibt Ereignisse und Werte zurück. Sie wird verwendet, um auf Benutzeraktionen wie das Klicken auf Schaltflächen oder das Schliessen des Fensters zu reagieren.
4. Openpyxl-Funktionen (`openpyxl.load_workbook()`, `workbook.active`, `worksheet.cell()`, `workbook.save()`): Diese Funktionen ermöglichen die Arbeit mit Excel-Dateien. Sie werden verwendet, um eine Excel-Arbeitsmappe zu öffnen, das aktive Arbeitsblatt, sowie Zellen im Arbeitsblatt basierend auf Zeile und Spalte auszuwählen und die Änderungen in der Arbeitsmappe zu speichern.

#### 5.5.4 Grasshopper: 3D-Modell Erstellung

In diesem Kapitel wird die Funktionsweise vom Grasshopper-Skript (Abb. 13) erklärt anhand der wichtigsten Abschnitte, die aus verschiedenen Nodes bestehen.

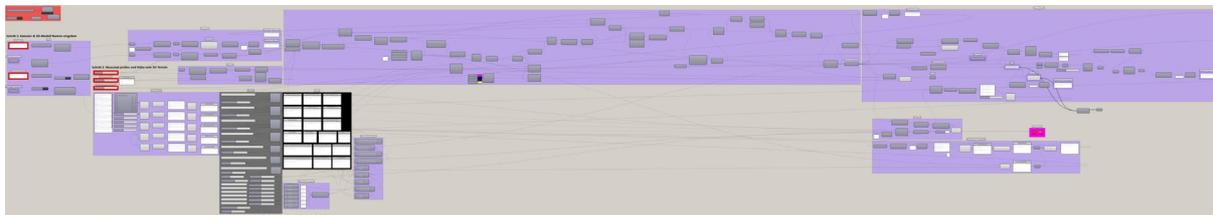


Abb. 13: Übersicht Grasshopper Skript

Es beginnt mit der „Input-Gruppierung“. Diese ist zuständig dafür, dass Kataster, Parzelle, 3D-Terrain und Excel Daten in Grasshopper eingelesen werden (Abb. 14). Die Python-Nodes sind zuständig dafür, dass sie die Eingabe vom Dateinamen aufgreifen und im Machbarkeitstool Order suchen. Die Nodes von der Parzelle und den Excel Dateien sind automatisch, weil die Namen von der Python-Anwendung vorgegeben werden und daher immer gleich bleiben.

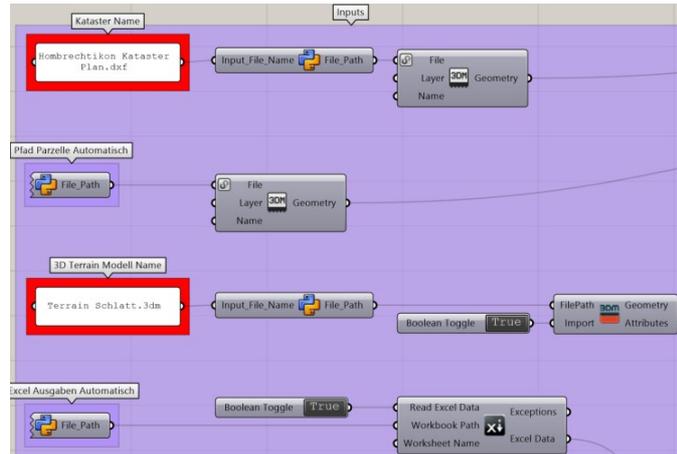


Abb. 14: Input Gruppierung

Weiter im Skript sind Einstellungsslider zu finden (Abb. 15). Der erste Slider ist da, um den Kataster und die Parzelle korrekt zu skalieren. Dieser muss veränderbar sein, da Katasterpläne nicht immer die gleichen Masstäbe haben. Der Zweite verschiebt das Terrainmodell entlang der Z-Achse auf den Nullpunkt von Rhino. Der letzte Slider verschiebt den Punkt, der massgebend ist für die Gebäudehöhe.

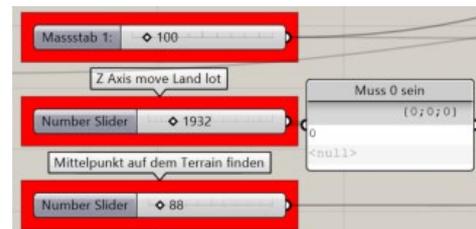


Abb. 15: Einstellungsslider

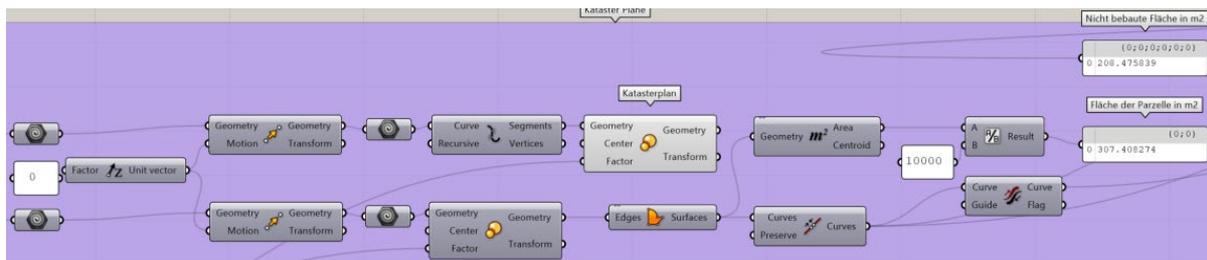


Abb. 16: Katasterplan Verarbeitung

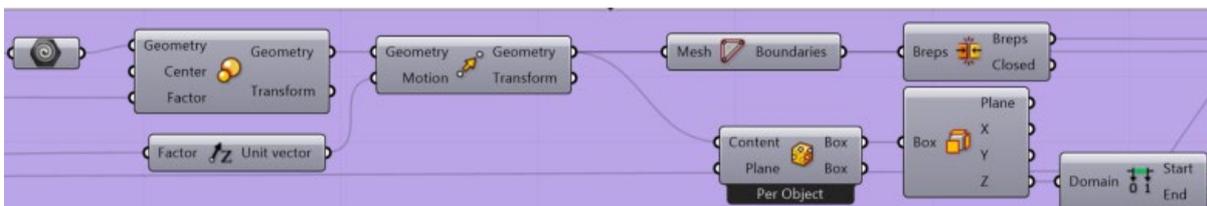


Abb. 17: 3D-Terrain Verarbeitung

Zusammenhängend mit dem Slider befinden sich Nodes, die die Informationen verarbeiten und den Geometrien weitergeben, damit sie in der 3D-Ansicht angezeigt werden (Abb. 16, 17). Im Kontext des Katasterplans (Abb. 16) wird dem Nutzer zusätzlich die Möglichkeit zur Korrektur bereitgestellt, indem die Quadratmeterzahl der jeweiligen Parzelle angezeigt wird. Dadurch erhält der Nutzer eine Rückmeldung über die Genauigkeit der Parzellengröße, mit der er arbeitet. Das Haus selbst wird aus der Parzelle extrudiert.

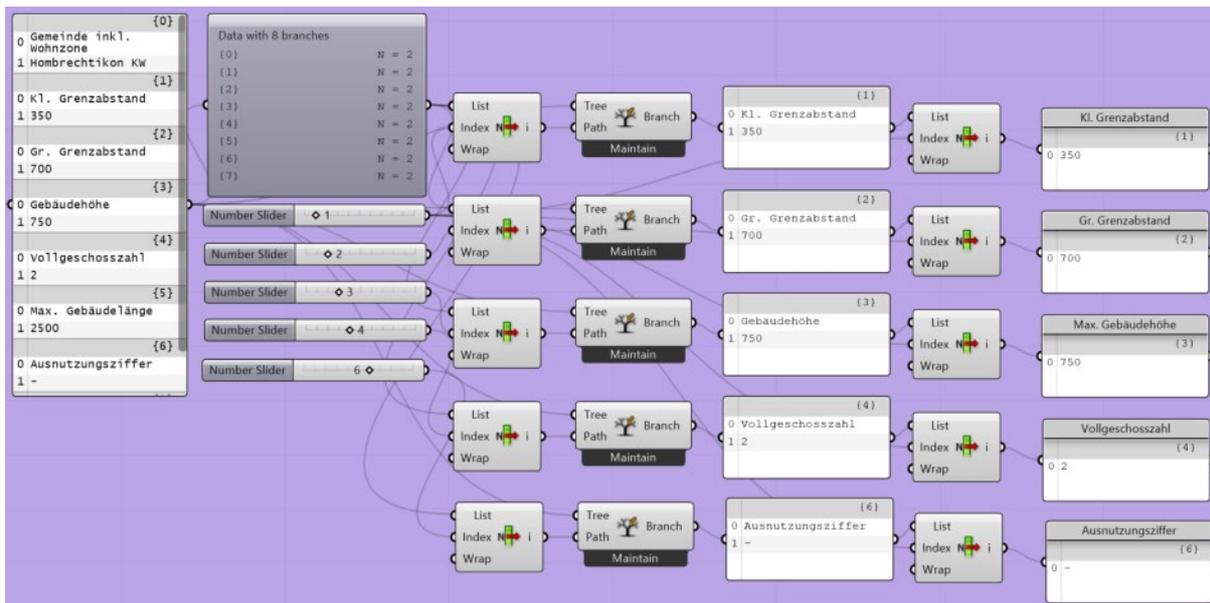


Abb. 18: Excel Aufspaltung

Ebenso muss das Excel auseinander genommen werden (Abb. 18), damit die Werte einzeln in den Inputs aufgenommen werden können. Vorausgesetzt der User hat zuvor die Bauvorschriften korrekt nach Vorlage in das Excel gespeichert. Falls dies der Fall ist, können die eingegebenen Vorgaben dann automatisch verwendet und im 3D-Modell implementiert werden.

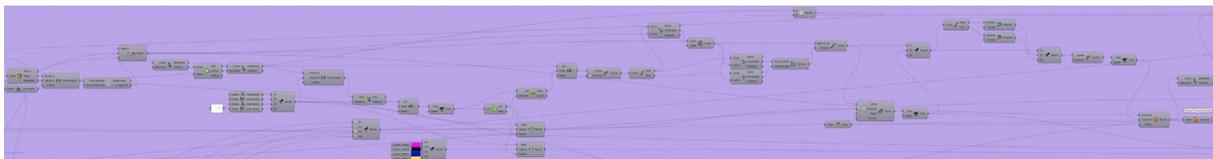


Abb. 19: Parzelle filtern nach N, O, S, W für Parzellengrenzen

Ein wichtiger Teil des Skripts beinhaltet die Erkennung der Kanten der Parzelle (Abb. 19). Das ist relevant, weil anschliessend die Filterung der Kanten nach Nord, Ost, Süd und West möglich ist, um danach die Versatzwerte (Parzellengrenzabstände) eingeben zu können.

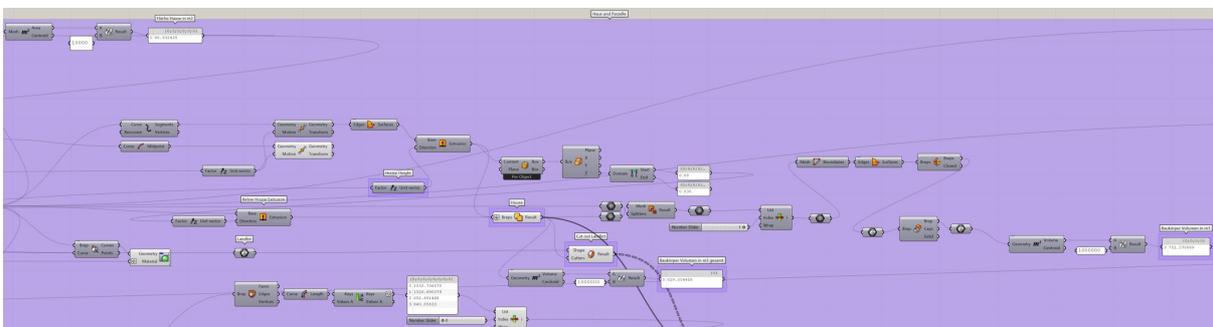


Abb. 20: Terrain / Haus Verschnitt

Diese Gruppierung ist zuständig, um Terrain und Haus miteinander zu verschneiden (Abb. 20). Die Verschneidung muss sauber verlaufen, da so die Fläche vom Haus nicht zu derjenigen der Umgebung gezählt wird. Anhand von dem können dann Aussagen über die Ausnutzung der Parzelle gemacht werden.

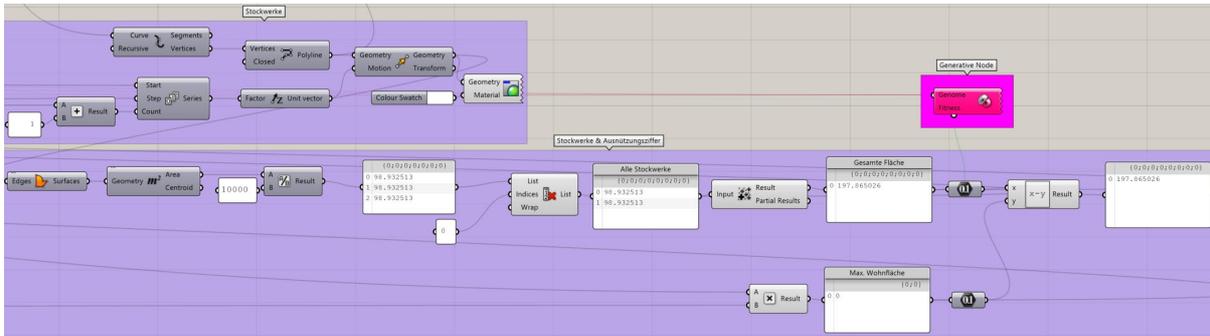


Abb. 21: Stockwerk Bearbeitung / Galapagos

Die Stockwerke leiten sich von der Grundfläche ab und werden entlang der Z-Achse nach oben kopiert. Es ist möglich das OK – OK Mass einzustellen sowie die Anzahl der Stockwerke. Das ermöglicht dem User im Zusammenhang mit der Höhe einzuschätzen, wie viele Stockwerke er bauen kann und wie hoch sie sein sollen. Im Skript (Abb. 21) gibt es eine Variante, mit dem Galapagos-Node (rosa) in Bezug auf die Ausnutzungsziffer ein Haus mit dem Parzellengrenzen-Slider zu generieren. Jedoch ist dies nicht zu empfehlen, da der Computer abstürzen könnte, falls er zu lange nach einer Lösung sucht, da der Solver keine Startwert Eingabe zulässt. Die erklärten Nodes sind wichtig zu verstehen, doch das, was für den Nutzer relevant ist, ist die grau eingefärbte Node-Gruppierung (Abb. 24). Sie beinhaltet alle Slider, die geändert werden können, um das Modell zu beeinflussen. Es sind alle mit dem angeschrieben, was sie ändern. Das Feedback im 3D geschieht synchron zu der Slider Änderung. Damit die Machbarkeitsstudie mit Werten beschrieben werden kann, folgen auf die Slider und deren Ausgabe einfache Gleichungen (Abb. 23, 25), die mit dem Output (Abb. 22) in Verbindung stehen. Der Output bildet das Fundament der Machbarkeitsstudie und macht die Visualisierung dazu noch verständlicher. Die Zahlen in den Output-Panels verändern sich gleichzeitig mit der Einstellung in der Steuerung (Abb. 24).

Fläche vom Haus in m2 0 98.932435	Fläche der Parzelle in m2 0 307.408274	Nicht ausgenutzt in % 0 67.81725
Nicht bebauete Fläche in m2 0 208.475839	Volumen Haus m3 0 753.578969	Ausnutzung in % 0 32.18275
Alle Stockwerke 0 98.932513 1 98.932513	Gesamte Fläche alle Stockwerke 0 197.865026	Max. Wohnfläche (AZ) 0 0
Gebäudehöhe 0 750	Gebäuelänge 0 1532.736578	Wlwg inkl. Konstruktionsflächen 0 193.75
Kosten/m3 Gebäudevolumen (GV) Norm SIA 416 0 534372.575859	Verfügbare Flächen (Nati. zu 0) 0 4.115026	Erthoffläche in m2 0 39.573005
Kosten/m2 Geschossfläche (GF) Norm SIA 416 0 414725.094496	Konstruktionsfläche in m2 0 38.75	Garagenplätze 0 2.967975

Abb. 22: Output Panel

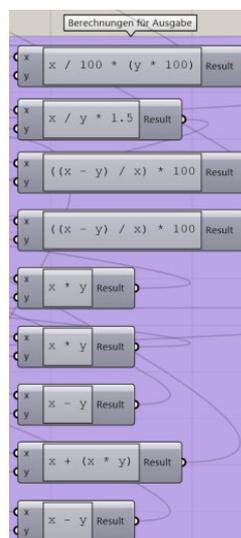


Abb. 23: Berechnung

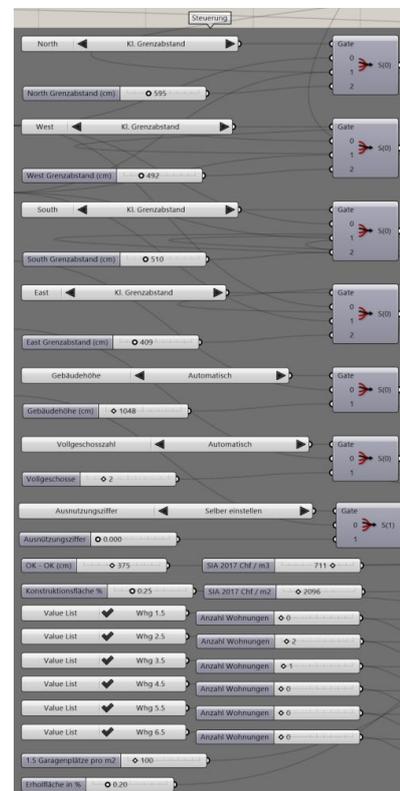


Abb. 24: Steuerung Slides

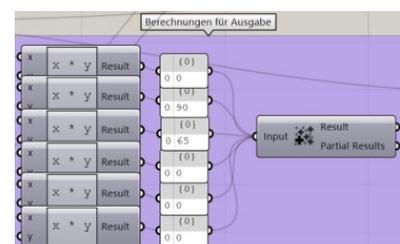


Abb. 25: Berechnung

## 6 Anwendung

### 6.1 Beschreibung des Anwendungsfalls

In diesem Abschnitt wird anhand vom entwickeltem Prototyp-Werkzeug im Kontext realer Parzellen aufgezeigt, wie der Workflow funktionieren könnte. Die Fallstudie konzentriert sich auf drei verschiedene Grundstücke in den Gemeinden Hombrechtikon, Lachen und Freienbach, welche jeweils alle eine andere Parzellenform aufweisen. Das Hauptziel der Fallstudie ist es, die Effektivität und Vielseitigkeit des Prototyps im Umgang mit unterschiedlichen Vorgaben zu demonstrieren.

- **Hombrechtikon** – Kt. Zürich – PLZ. 8634 – Gesamtfläche 14.5 km<sup>2</sup>
- **Lachen** – Kt. Schwyz – PLZ. 8853 – Gesamtfläche 2.4 km<sup>2</sup>
- **Freienbach** – Kt. Schwyz – PLZ. 8807 – Gesamtfläche 13.6 km<sup>2</sup>

Im Rahmen der Fallstudie wird der Prototyp des Tools auf jede der drei Parzellen angewendet, um seine Anpassungsfähigkeit an unterschiedliche Kontexte zu beschreiben. Darüber hinaus soll das Potenzial des Prototyps für eine fundiertere Entscheidungsfindung aufgezeigt werden.

### 6.2 Datenintegration und Aufbereitung

Zuerst mussten die Parzellen definiert werden, die hier bearbeitet werden sollten. Sie wurden möglichst beliebig gewählt in unterschiedlichen Bauzonen, jedoch wurde sich auf Wohnbauten konzentriert. Das ist wichtig, um zu zeigen, dass nichts an der Logik vom Prototypen verändert werden muss, um mit unterschiedlichen Vorgaben arbeiten zu können. Die Katasterpläne wurden vom GIS-Browser vom Kanton Schwyz und Zürich heruntergeladen. Da die Katasterpläne keine Vektordateien waren, mussten sie nachgezeichnet werden und die Parzelle, die bearbeitet werden soll, rot eingefärbt (Abb. 26, 27, 28). Das ist relevant, um dem Python Skript sagen zu können, welche Parzellengrenzen ausgewählt werden sollen.

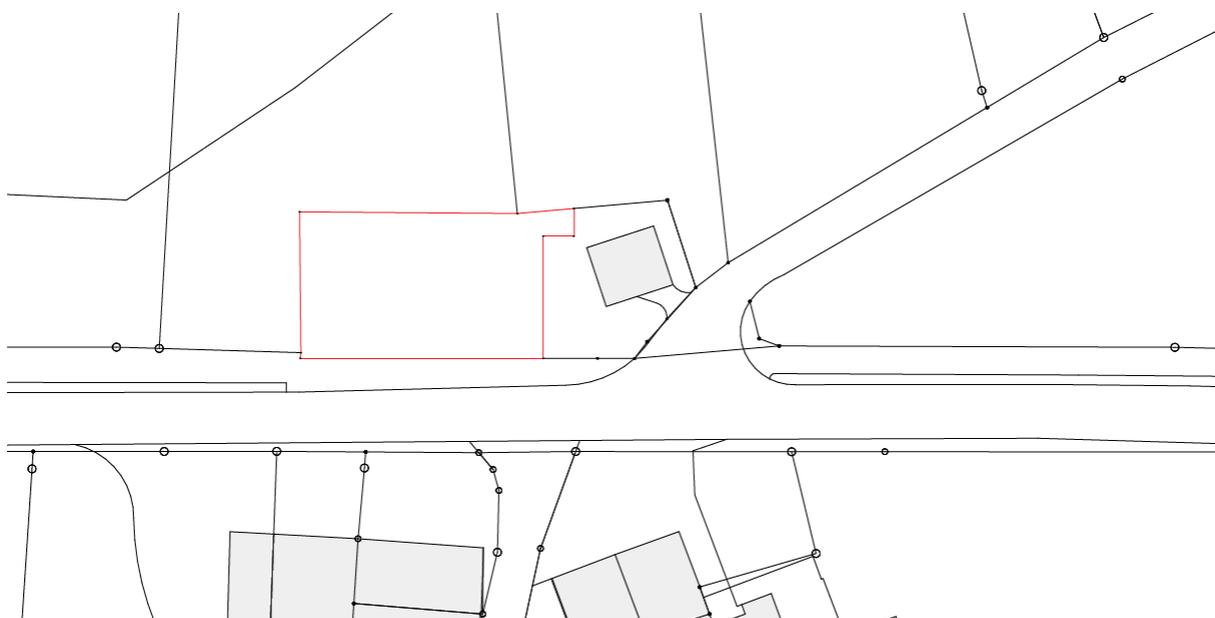


Abb. 26: Katasterplan Hombrechtikon KW (nicht massstäblich, wahre Grösse der Parzelle 307m<sup>2</sup>)

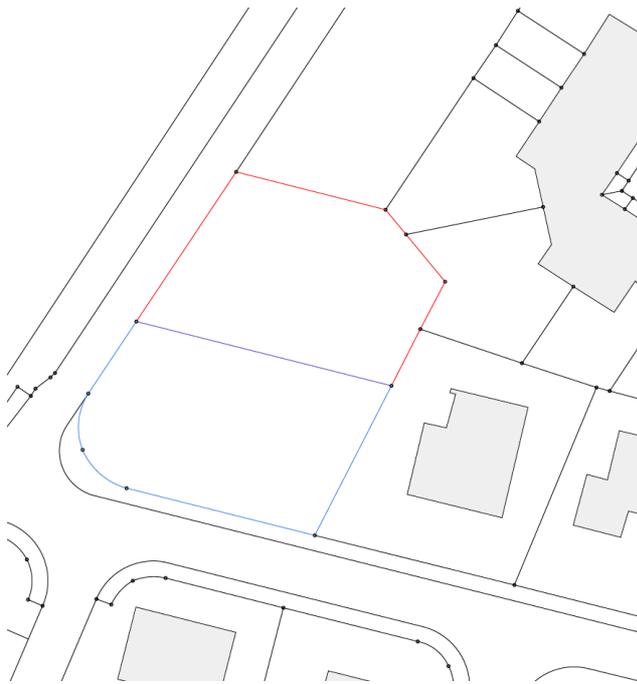


Abb. 27: Katasterplan Lachen W2 (nicht massstäblich, wahre Grösse der Parzelle 657m<sup>2</sup>)

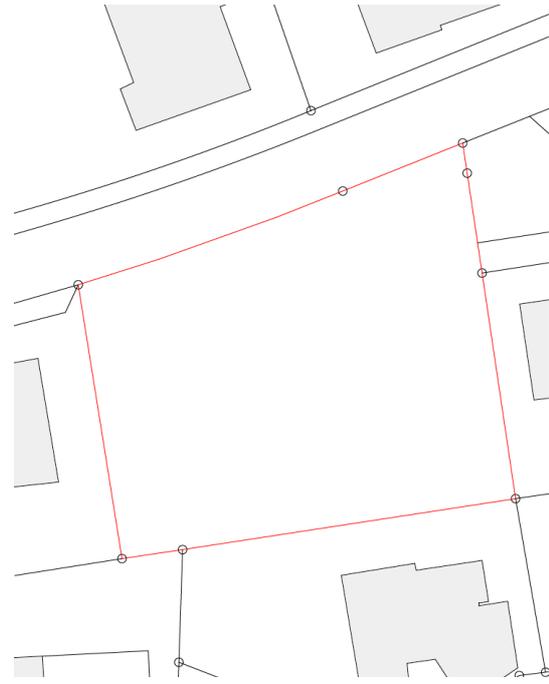


Abb. 28: Katasterplan Freienbach W3 (nicht massstäblich, wahre Grösse der Parzelle 1092m<sup>2</sup>)

Als Nächstes mussten die Daten vom Baureglement strukturiert werden, damit sie vom Grasshopper verwendet werden können. Dies wurde mit dem Tool gemacht, das im Kapitel 5.5.3 beschrieben wurde, dass die Vorgaben in eine zentrale Excel Datei schreibt (Abb. 29). So wurde Hombrechtikon KW, Lachen W2 und Freienbach W3 abgespeichert (Abb. 30).

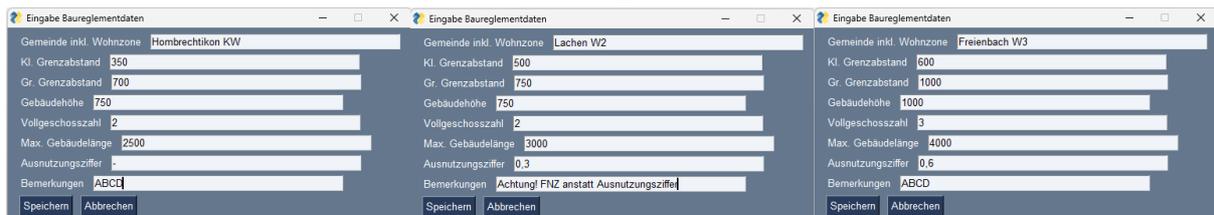


Abb. 29: Eingabe Bauvorgaben

Gemeinde inkl. Wohnzone	Kl. Grenzabstand	Gr. Grenzabstand	Gebäudehöhe	Vollgeschosszahl	Max. Gebäudelänge	Ausnutzungsziffer	Bemerkungen
Hombrechtikon KD	250,00	400,00	1400,00	2,00	3500,00	0,60	ABCD
Hombrechtikon KW	350,00	700,00	750,00	2,00	2500,00	-	ABCD
Hombrechtikon Z3/50	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon W1/20	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon W2/30	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon W2/35	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon W3/55	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon WG2/35	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Hombrechtikon WG3/55	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Lachen K1	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen K2	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen KR	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen W1	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen W2	500,00	750,00	750,00	2,00	3000,00	0,30	Achtung! FNZ anstatt
Lachen W3	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen WG2	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Lachen WG3	250,00	400,00	1400,00	2,00	3000,00	0,60	ABCD
Freienbach K	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Freienbach Z	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Freienbach W4	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Freienbach WG3	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Freienbach W2	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD
Freienbach W3	600,00	1000,00	1000,00	3,00	4000,00	0,60	ABCD
Freienbach W4	250,00	400,00	1400,00	3,00	3000,00	0,60	ABCD

Abb. 30: Bauvorgaben Excel

Zuletzt muss noch ein Höhenmodell der jeweiligen Parzellen erhoben werden. Es gibt verschiedene Daten, die auf der Swisstopo Seite zu finden sind. Doch diese 3 Gemeinden sind noch nicht auf einem genügenden Stand. Die Terrainmodelle wurden daher von Hand im ArchiCAD erstellt anhand der Höhenlinien, die im GIS-Browser zu finden sind (Abb. 31).

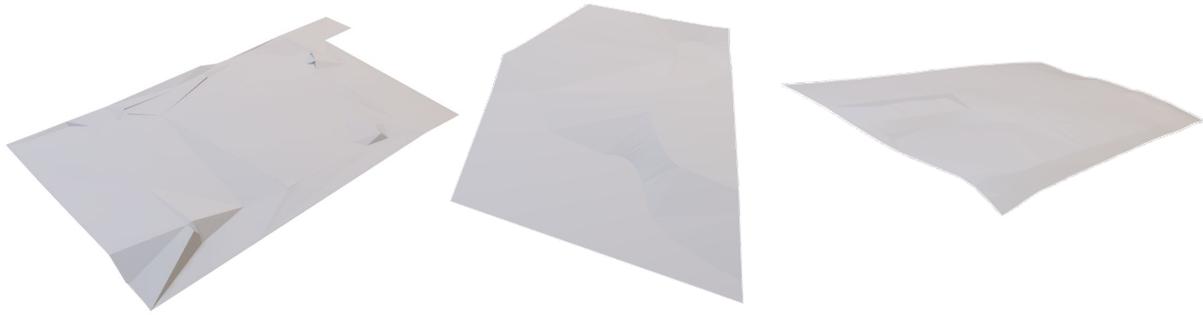


Abb. 31: 3D-Terrain Modelle (Hombrechtikon, Lachen, Freienbach)

Ein zusätzliches Auswahlkriterium betraf die Steigung der Topologie. Im Gegensatz zu Hombrechtikon und Lachen, ist die Parzelle in Freienbach eher steil mit einem Höhenunterschied von 6.2 m. Sobald diese drei Informationen zusammengetragen wurden, kann mit dem Prozess der Erstellung vom 3D-Modell begonnen werden.

### 6.3 Zusammenführung

In dieser Phase wird die in Abschnitt 5.5.2 dargelegte Anwendung angewendet, um Katasterpläne und baurechtliche Bestimmungen in die Grasshopper-Datei zu integrieren. Da dieser Schritt für alle Grundstücke identisch ist, mit Ausnahme der verschiedenen Excel-Ausgaben, wird er hier nur am Beispiel des Grundstücks Hombrechtikon KW die Anwendung gezeigt. Das Tool muss wie schon erklärt aus dem Ordner auf dem Desktop aufgemacht werden (Abb. 32).

Baureglement daten Input in Excel.exe	05/05/2023 19:23	Application	35.788 KB
<input checked="" type="checkbox"/> Katasterplan Analyse Anwendung.exe	05/05/2023 19:15	Application	84.317 KB
Hombrechtikon Kataster Plan.dxf	08/03/2023 10:51	DXF File	1.493 KB
Parzelle gefiltert.dxf	06/05/2023 14:04	DXF File	20 KB
Prototype Katasterplan.gh	06/05/2023 11:36	Grasshopper Definiti...	115 KB
Excel Baureglementdaten.xlsx	06/05/2023 12:16	Microsoft Excel Work...	12 KB
Excel BaureglementdatenOutput.xlsx	06/05/2023 14:05	Microsoft Excel Work...	5 KB
Terrain Schlatt.3dm	08/03/2023 15:34	Rhino 3-D Model	19 KB

Abb. 32: Prototyp Tool Ordner

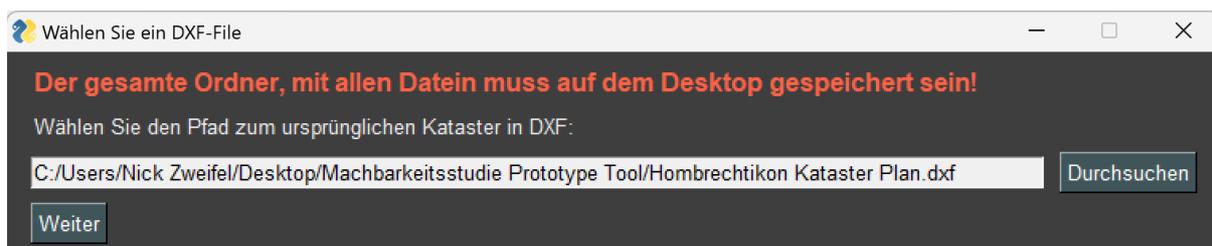


Abb. 33: Schritt 1 der Anwendung

Der Benutzer wird dann anhand der Popup-Fenster durch einen vordefinierten Ablauf geführt. Zu Beginn muss der Pfad vom Katasterplan in Hombrechtikon ausgewählt werden (Abb. 33).



Abb. 34: Verschiedenen Farben im DXF

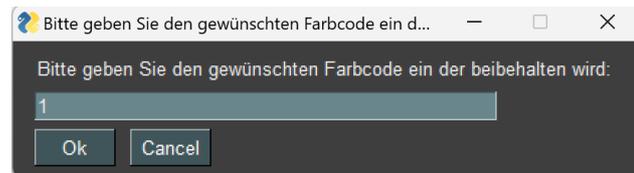


Abb. 35: Farbe angeben

Die Farbinformationen im Katasterplan werden im Folgenden analysiert. Da das DXF-Dateiformat von Autodesk entwickelt wurde, enthält es spezifische numerische Codes zur Beschreibung von Farben. Im Katasterplan von Hombrechtikon sind die Farbcodes 1 und 7 repräsentiert, entsprechend den Farben Rot und Schwarz (Abb. 34). Da die Parzellengrenzen zuvor in Rot dargestellt wurden, wird hier der Farbcode 1 verwendet (Abb. 35). Es ist anzumerken, dass die Grenzen nicht zwingend rot sein müssen, sondern lediglich von den übrigen Elementen farblich abweichen sollten. Infolgedessen wird in diesem Schritt automatisch eine neue DXF-Datei generiert, die nur die betreffende Parzelle abbildet.

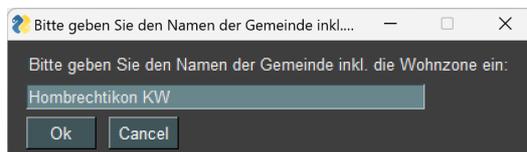


Abb. 36: Gemeinde Abfrage



Abb. 37: Gemeindeinformationen Ausgabe

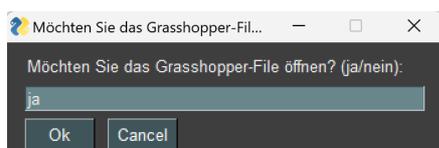


Abb. 38: Grasshopper Öffnung

Danach muss die Gemeinde und die Wohnzone eingegeben werden, damit die richtigen Informationen in das Grasshopperfile gelangen (Abb. 36). Das hängt zusammen mit dem zuvor beschriebenen Excel. Als Kontrolle werden sie nochmals zur Veranschaulichung visualisiert (Abb. 37). Zuletzt wird gefragt, ob das Grasshopperfile im Ordner geöffnet werden soll, für die 3D-Modell Generierung (Abb. 38).

## 6.4 Erstellung von 3D-Modellen

In diesem Kapitell erfolgt die Kombination der zwei vorherigen Schritte, die zu einem Ganzen zusammengefügt werden. Da sich die Schritte in diesem Kapitel bei allen drei Parzellen wiederholen, wird anhand vom Beispiel in Hombrechtikon die Erstellung gezeigt. Die Modelle und Outputs von Lachen und Freienbach werden am Ende gezeigt.

Der erste Schritt, der durchgeführt werden muss, ist die ordnungsgemäße Verknüpfung vom gesamten Katasterplan und dem Terrainmodell. Hierzu müssen in die weissen Felder die Namen mit Dateierweiterung geschrieben werden (rot markiert) (Abb. 39). Auf diese Weise kann das Skript die korrekten Dateien identifizieren, die in das Skript geladen werden sollen.

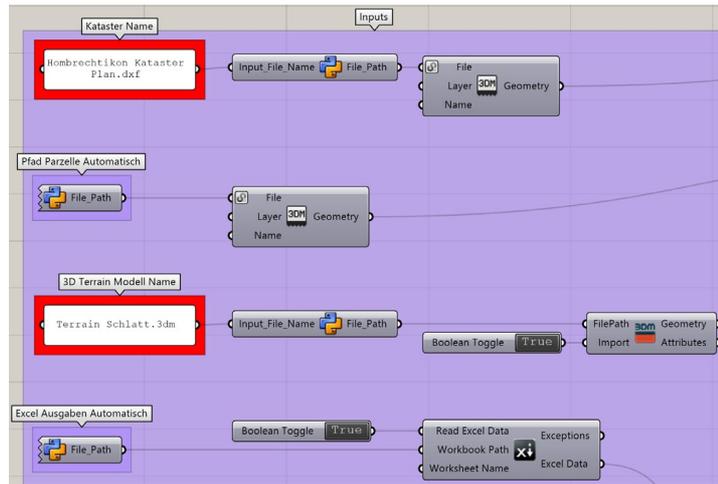


Abb. 39: Namen eingeben

Im Anschluss muss die Z-Achse des Terrains angepasst werden, sodass der unterste Punkt des Modells auf demselben Punkt liegt, wie der Kataster (Abb. 40). Bei der Parzelle muss das nicht gemacht werden, weil die gefilterte Parzelle eine Kopie vom Kataster ist, somit stimmen die Höhen überein. Auch der Masstab muss noch korrekt angegeben werden, damit der Kataster richtig skaliert werden kann und die Masse schlussendlich stimmen. In diesem Beispiel war der Kataster im Masstab 1:100 (Abb. 41).



Abb. 40: Z-Achse Slider



Abb. 41: Masstab Änderung

Folgend kann das 3D-Modell angepasst werden. Wie im vorherigen Kapitel 5.5.4 beschrieben, muss ab da mit der Steuerung gearbeitet werden, welche sich von oben nach unten liest (Abb. 43). Zunächst sollten die Grenzabstände festgelegt werden. In diesem Beispiel handelt es sich um ein reales Projekt, dass gebaut wurde. Daher gibt es keinen grossen Grenzabstand von 7 m, sondern vier kleine Grenzabstände von je 3,5 m. Dies liegt daran, dass sich im Süden eine Strasse befindet und der Strassenabstand 3,5 m beträgt, die Gemeinde war damit einverstanden. Ein Grenzabstand von 7 m würde die Nutzung von der Parzelle dieser Grösse erheblich einschränken.

Die Gebäudehöhe leitet sich vom Schnittpunkt (in der Abb. 43 blau) Mitte der Fassade und massgebendem Terrain ab. Auch diese Vorschrift ist in verschiedenen Gemeinden anders definiert. Das Grasshopper Skript erlaubt es, diesen Punkt manuell auf der Z-Achse zu verschieben (Abb. 42).



Abb. 42: Mittelpunkt verschieben

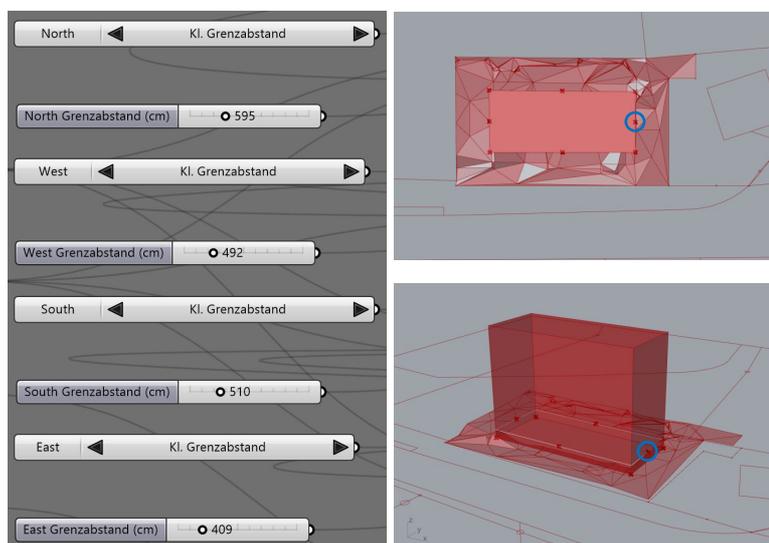


Abb. 43: Steuerung und korrespondierendes Modell

Im unteren Bereich der Steuerung können weiterführende Informationen ergänzt werden (Abb. 44). Bei Aspekten wie Gebäudehöhe, Vollgeschosszahl und Ausnutzungsziffer (AZ) besteht die Option, entweder automatisch die Werte aus der Excel-Tabelle zu verwenden oder manuell mithilfe eines Sliders anzupassen. Hier in diesem Beispiel ist die AZ auf 0 gesetzt. Auch können die Wohnungen eingestellt werden, mit Hilfe vom durchschnittlichen Flächenzahlen. Auf diese Weise lässt sich abschätzen, wie viele Wohneinheiten in dem Gebäude untergebracht werden könnten. Neben der Steuerung befindet sich das Output-Dashboard (Abb. 47). Ein wesentlicher Vorteil darin, ist, dass sofort Feedback im Output erhalten wird, wenn die Slider geändert werden. Zum Dashboard kann eine 3D-Datei erstellt werden, die das Dashboard greifbarer macht (Abb. 45, 46).

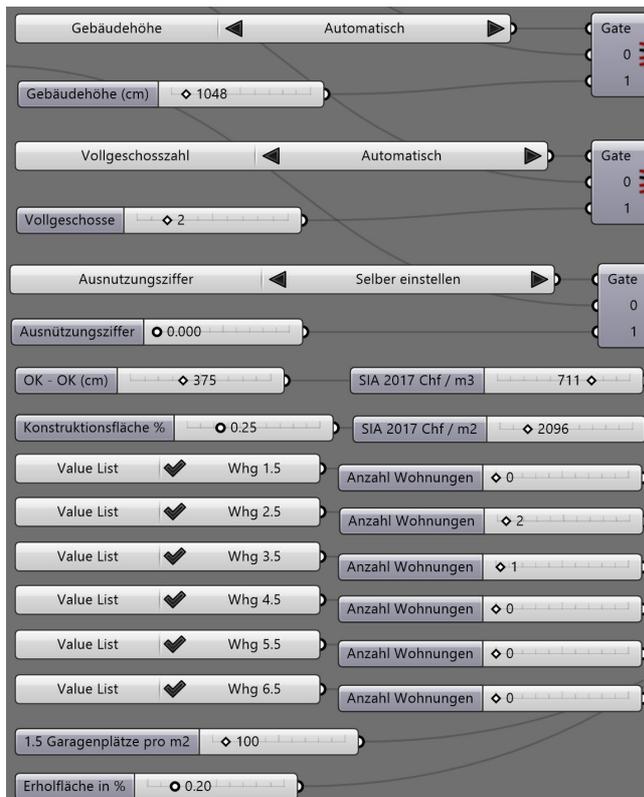


Abb. 44: Steuerung unterer Teil

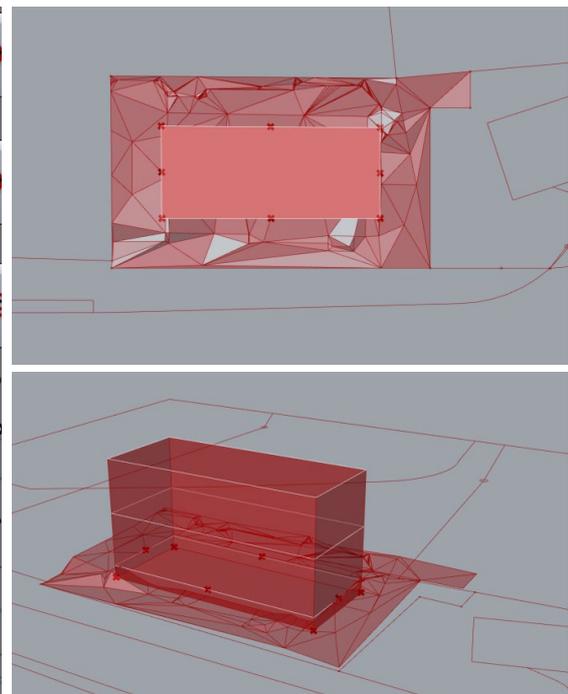


Abb. 45: Modell Grundriss und Ansicht

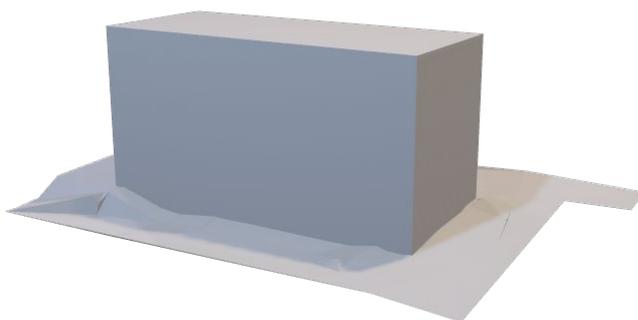


Abb. 46: STL-Modell Hombrechtikon

Fläche vom Haus in m2 (0;0;0;0;0;0;0)	Fläche der Parzelle in m2 (0;0;0;0;0;0;0)	Nicht ausgenutzt in % (0;0;0;0;0;0;0)	
0 98.932435	0 307.408274	0 67.81725	
Nicht bebaute Fläche in m2 (0;0;0;0;0;0;0)	Volumen Haus m3 (0;0;0;0;0;0;0)	Ausnutzung in % (0;0;0;0;0;0;0)	
0 208.475939	0 751.578869	0 32.18275	
Alle Stockwerke (0;0;0;0;0;0;0)	Gesamte Fläche alle Stockwerke (0;0;0;0;0;0;0)	Max. Wohnfläche (AZ) (0;0;0;0;0;0;0)	
0 98.932513 1 98.932513	0 197.865026	0 0	
Gebäudehöhe (0;0;0;0;0;0;0)	Gebäuelänge (0;0;0;0;0;0;0)	Whg inkl. Konstruktionsflächen (0;0;0;0;0;0;0)	Erholfläche in m2 (0;0;0;0;0;0;0)
0 750	0 1532.736578	0 193.75	0 39.573005
Kosten/m3 Gebäudevolumen (GV) Norm SIA 416 (0;0;0;0;0;0;0)	Verfügbare Flächen (Nah zu 0) (0;0;0;0;0;0;0)	Whg exkl. Konstruktionsflächen (0;0;0;0;0;0;0)	
0 534372.578859	0 4.115026	0 155	
Kosten/m2 Geschossfläche (GF) Norm SIA 416 (0;0;0;0;0;0;0)	Konstruktionsfläche in m2 (0;0;0;0;0;0;0)	Garagenplätze (0;0;0;0;0;0;0)	
0 414725.094496	0 38.75	0 2.967975	

Abb. 47: Output Panel





## 6.5 Implementierte Informationen im Output

Dieses Kapitel dient zur Ergänzung zum Vorherigen, warum jede der folgenden Output Informationen im Zusammenhang mit dem 3D-Modell ausgegeben wird:

**Fläche vom Haus in m<sup>2</sup>:** Zeigt die Grundfläche des Gebäudes an, wichtig für die Planung und Berechnung der Baukosten.

**Fläche der Parzelle in m<sup>2</sup>:** Gibt den verfügbaren Platz für das Gebäude an und hilft bei der Einhaltung von Grenzabständen und Bebauungsvorschriften.

**Nicht ausgenutzte Fläche in % und m<sup>2</sup>:** Zeigt die noch verfügbare Baufläche an, die nicht vom Haus beansprucht wird.

**Volumen vom Haus in m<sup>3</sup>:** Hilft bei der Bestimmung von Raumkapazitäten und der Einschätzung von Baukosten und Materialbedarf.

**Ausnutzung in %:** Zeigt, wie effizient der verfügbare Raum genutzt wird, und kann zur Optimierung des Gebäudedesigns verwendet werden.

**Fläche aller Stockwerke in m<sup>2</sup>:** Gibt Aufschluss über die Gesamtfläche des Gebäudes, wichtig für die Planung von Innenräumen.

**Max. Wohnfläche (AZ) in m<sup>2</sup>:** Gibt an, wie viel Wohnfläche laut Vorschriften auf der Parzelle zulässig ist. Damit das Gebäude noch passend angepasst werden kann.

**Gebäudehöhe und -länge:** Wichtig für die Einhaltung von Bauvorschriften und die Integration des Gebäudes in die Umgebung.

**Whg inkl. Konstruktionsflächen in m<sup>2</sup>:** Zeigt die Wohnfläche inklusive Pauschale für Wände und Konstruktionsflächen an, wichtig für die Bestimmung des Raumbedarfs.

**Erholungsfläche in m<sup>2</sup>:** Gibt an, wie viel Platz für Erholungsbereiche eingeplant werden muss, im Zusammenhang der Ausnutzung.

**Kosten / m<sup>3</sup>:** Hilft bei der Budgetierung und Einschätzung der Baukosten.

**Verfügbare Fläche für Whg (muss nahezu 0 sein):** Zeigt, ob die gesamte erlaubte Wohnfläche ausgeschöpft wurde.

**Whg exkl. Konstruktionsflächen in m<sup>2</sup>:** Gibt die tatsächliche Wohnfläche ohne Wände und Konstruktionsflächen an, wichtig für die Planung von Innenräumen.

**Kosten / m<sup>2</sup>:** Ermöglicht einen Vergleich der Baukosten zwischen verschiedenen Projekten und Standorten.

**Konstruktionsfläche in m<sup>2</sup>:** Zeigt die Fläche, die von tragenden Wänden und Strukturen eingenommen wird.

**Garagenplätze:** Gibt die Anzahl Parkplätze an, die im Zusammenhang mit der Wohnfläche geplant werden müssen, wichtig für die Einhaltung von Parkvorschriften und die Planung des Verkehrsflusses.

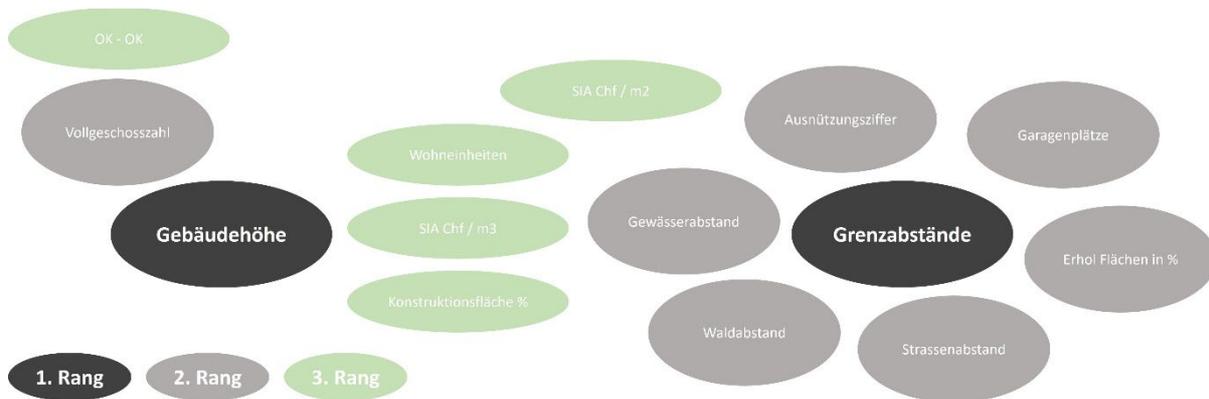


Abb. 56: Veranschaulichung der Slider

Die Grafik (Abb. 56) zeigt die Hierarchie der Slider und wo sie sich zwischen den anderen befinden. Die Wichtigsten und die, welche alles beeinflussen, sind die Grenzabstände und die Gebäudehöhe. Der Grenzabstand Slider kann die im Baureglement vorgegebenen Strassen, Wald und Gewässerabstände beeinflussen, da diese Abstände mehrheitlich von der Grenze der Parzelle ausgehen. Die AZ wird nicht direkt vom Grenzabstandsslider gesteuert, da es sich um einen fixen Wert handelt. Vielmehr dient dieser als Richtlinie für die Grenzabstände. Dies deshalb, weil sich mit deren Änderung die gesamte  $m^2$  vom Haus verkleinert oder vergrössert und sich so an die Vorgabe der AZ genähert oder entfernt wird. Auch hängen die Garagenplätze und die Erholungsflächen, die zu planen sind, mit der Grösse vom Haus zusammen und sind somit wiederum mit den Grenzabständen verknüpft.

Die grünen Formen repräsentieren variable Slider, die im 3. Rang der Hierarchie angesiedelt sind. Diese Slider ermöglichen eine feinere Anpassung und Optimierung der Planungsparameter, um den individuellen Anforderungen und Präferenzen gerecht zu werden. Die 3-rangige Hierarchie gewährleistet, dass die grundlegenden Aspekte der Planung wie Gebäudehöhe und Grenzabstände (1. Rang) zunächst berücksichtigt werden, bevor weitere Parameter wie AZ, Garagenplätze und Erholungsflächen (2. Rang) angepasst werden. Schliesslich ermöglichen die 3. Rang Slider die Feinjustierung der Planung, um optimale Ergebnisse zu erzielen und individuellen Bedürfnissen gerecht zu werden.

## 6.6 Fazit der Erstellung

Um die drei Ergebnisse zu erreichen, wie im Kapitel 6.4 gezeigt, wurden 20 Minuten aufgewendet. Hier wurde die Zeit für die Analyse der Baureglemente nicht in Betracht gezogen. Aus der Anwendung auf diese drei Parzellen ging hervor, dass die Effizienz zum einen mit dem Tool zu tun hat, aber auch wie Dateien wie Katasterplan, Terrainmodell und Baureglemente zu Beginn erstellt wurden. Die Gemeinde Lachen zum Beispiel verweist an einigen Stellen im Reglement auf das PBR. Das hat zur Folge, dass einige Stellen zu einem gewissen Grad auf mehrere Weise interpretiert werden können und Zeit aufgewendet werden muss, um die ergänzenden Informationen zu suchen. Im Vergleich macht das die Gemeinde Freienbach deutlich besser, da sie die ausschlaggebenden Vorschriften wie Parzellengrenzen und Ausnutzungsziffer pro Zone in eine Tabelle geschrieben haben. Aus diesem resultiert, dass die Reglemente die ersten und wichtigsten Herausforderungen bei der Erstellung sind.

Um einen Vergleich zu schaffen, dauert eine Machbarkeitsstudie, wie im Kapitel 2.2 beschrieben wurde, 3-6 Monate. Spezifisch, die technische Machbarkeit dauert bis zu 3 Wochen. Wenn als Beispiel von 2 Wochen ausgegangen wird (80h) für die Parzelle Hombrechtikon, dann wird mit dem Prototyp von die benötigte Zeit um 98 % reduziert, wenn der Benutzer 1 Stunde hat inkl. Analyse vom Baureglement und Erstellung vom Modell. Eine vollständige Machbarkeitsstudie liegt jedoch nach dieser Stunde noch nicht vor, doch eine erheblich schneller erarbeitete Grundlage für weitere Analysen der Wirtschaftlichkeit und Gesellschaft. Da diese immer zuerst von den technischen Aspekten ausgehen.

## 6.7 Abgrenzung

Der in dieser Studie vorgestellte Prototyp wurde entwickelt, um die Planung von Gebäuden unter Berücksichtigung verschiedener Parameter wie Gebäudehöhe, Grenzabstände und Bauvorschriften zu optimieren. Obwohl sich der Prototyp als nützliches Werkzeug erwiesen hat, ist es wichtig, seine Grenzen und die Bereiche, die bei seiner Entwicklung nicht berücksichtigt werden konnten zu kennen.

Eine wesentliche Einschränkung des Prototyps besteht darin, dass er die Form eines komplizierteren Grundstücks mit sieben oder acht Seiten nicht immer genau erfassen kann. Dieses Problem ergibt sich aus der Schwierigkeit zu bestimmen, welche Parzellengrenzlinie welcher Richtung entspricht. Zukünftige Iterationen des Prototyps sollten dieses Problem angehen, indem ein Algorithmus entwickelt wird, der mit komplexen Parzellenformen umgehen kann und der die Seiten genau den entsprechenden Richtungen zuordnet. Von Hand wäre es kein Problem, doch wie erläutert liegt das Potenzial in der Automation. Doch dies zu automatisieren, erweist sich als erheblich schwieriger als angenommen.

Ein weiterer Aspekt, der im aktuellen Prototyp nicht berücksichtigt wird, ist die Implementierung eines Algorithmus, der Gebäude, nach einer bestimmten Länge, teilt. Derzeit erlaubt der Prototyp dem Benutzer nur die Erstellung einer einzelnen Form innerhalb der Parzellengrenzen und die Ableitung von Informationen auf Basis dieser Form. Die Integration eines Algorithmus, der Gebäude nach einer bestimmten Länge, zum Beispiel die max. Gebäudelänge teilt, würde die Möglichkeiten des Prototyps erweitern. Das Tool könnte dann auch für grössere Parzellen benutzt werden, um Areale mit mehreren Gebäuden zu planen. Ausserdem konzentriert sich der Prototyp ausschliesslich auf das oberirdische Volumen. Die Untergeschosse sind zwar baurechtlich zu beurteilen, doch eine überwiegende Anzahl von Vorschriften betreffen nur das oberirdische Volumen. Darum wurde auch für die drei Parzellen das unterirdische Volumen nicht berücksichtigt, da keine relevanten Themen, die auf diese Parzellen Grössen anzuwenden sind, in den Baureglementen beschrieben sind. Wenn das unterirdische Volumen im Skript berücksichtigt wird, können auf grossen Überbauungen Aussagen über Tiefgaragenplätze gemacht werden.

Zusammenfassend kann gesagt werden, dass der in dieser Studie vorgestellte Prototyp ein wertvolles Werkzeug für die Optimierung von Gebäudeentwürfen ist. Durch die Beseitigung dieser Einschränkungen und die Integration zusätzlicher Funktionen kann der Prototyp weiter verbessert und zu einem noch wertvolleren Werkzeug für Planer und Entscheidungsträger im Bereich der Planung werden.

## 7 Diskussion

### 7.1 Zusammenfassung der Arbeit

In der Arbeit wurde untersucht, wie digitale Möglichkeiten das Potenzial haben die Durchführung von Machbarkeitsstudien effizienter zu gestalten. Die Forschungsthese konnte bestätigt werden, indem ein Prototyp mit Hilfe von Python und Grasshopper erstellt wurde. Zusätzlich zu der Erstellung wurde gezeigt, wie die Verwendung funktioniert und was daraus resultiert. Der Inhalt der Arbeit gliedert sich in mehrere Abschnitte, die zuerst verschiedenartige Aspekte von Machbarkeitsstudien und den Einsatz digitaler Technologien abdecken. Zunächst wurden die Grundlagen von Machbarkeitsstudien, aber auch die Rollen, die sie spielen, erörtert. Anschliessend wurde die Bedeutung von Effizienz in der Bauindustrie und die Rolle von digitalen Technologien untersucht. Bestehende Tools wie Sidewalk Labs, Spacemaker und Archistar wurden analysiert, um Erkenntnisse für die Entwicklung eines eigenen digitalen Tools zu gewinnen. Auch dient die Analyse dazu, Probleme im Zusammenhang mit der vollständigen Automatisierung zu erkennen und zu verstehen.

Der Hauptteil der Arbeit befasst sich mit der Entwicklung eines Prototyps, der die Machbarkeitsstudie digital unterstützen kann, einschliesslich der Auswahl und Begründung der eingesetzten Technologien, der Datenintegration und Aufbereitung sowie der automatisierten Zusammenführung und Erstellung von 3D-Modellen. Erklärungen und Beschreibungen der Skripte in Python und Grasshopper wurden vorgestellt, damit die Möglichkeit besteht in Zukunft Weiterführungen vom Prototyp aus machen zu können. Die Anwendung des entwickelten Prototyps wurde anhand von konkreten Anwendungsfällen in Hombrechtikon, Lachen und Freienbach demonstriert. Die Implementierung von Informationen im Output und die Abgrenzung der Ergebnisse wurden ebenfalls diskutiert.

### 7.2 Beantwortung der Forschungsfrage

Die gesamte Arbeit hat gezeigt, dass eine Automatisierung mit Hilfe von digitalen Mitteln in der Durchführung von Machbarkeitsstudien ein erhebliches Potenzial birgt. Es muss dennoch gesagt werden, dass viel auch davon abhängt, wie gut strukturiert die verfügbaren Daten sind. Wie in der Arbeit beschrieben wurde, ist es nötig, dass alle Bauvorschriften zentral vorhanden und im gleichen Format sind. Somit geschehen weniger Fehler und die Informationen können immer gleich aufgegriffen werden, wie anhand vom Beispiel in Grasshopper vorgestellt wurde. Es ist schwierig, eine Prozentzahl zu nennen, welche die Effizienz im Vergleich zur herkömmlichen Machbarkeitsstudie beschreibt. Jedoch kann gesagt werden, dass für die drei Parzellen für die Erstellung 20 Minuten aufgewendet wurde und die Zeit für die Prüfung der technischen Machbarkeit um 98 % reduziert werden konnte. Daraus geht hervor, dass mit dem Tool erheblich Zeit gespart wird, vor allem in Bezug auf die Erstellung von 3D-Modellen und den vorgestellten Outputs. Darüber hinaus ermöglicht es eine schnellere Entscheidungsfindung.

Letztendlich lässt sich sagen, dass der entwickelte Prototyp und die zugrunde liegende Methodik den Anfang eines langen Weges darstellen, um die Durchführung von Machbarkeitsstudien zu optimieren.

Es ist dementsprechend wichtig, dass das Tool Überschreitungen zulässt, um baurechtliche Grauzonen zu finden, damit das Mögliche auf einer Parzelle noch weiter ausgereizt werden kann. Durch zusätzliche Forschung und Entwicklung in diesem Bereich kann die Effizienz und Genauigkeit von Machbarkeitsstudien optimiert werden.

### 7.3 Empfehlung für zukünftige Forschungen

Die Ergebnisse dieser Arbeit legen nahe, dass digitale Technologien das Potenzial haben, Machbarkeitsstudien in der Bauindustrie effizienter zu gestalten. Zukünftige Forschungen sollten zu Beginn darauf abzielen, die Benutzerfreundlichkeit des entwickelten Tools zu verbessern. Dazu gehört die Übersetzung in eine andere Sprache, damit der User keine Grasshopper Lizenz besitzen muss. Eine mögliche Variante wäre alles, was in diesem Prototyp mit Grasshopper gemacht wurde, mit Python zu schreiben.

Darüber hinaus ist es wichtig, die Interaktion und Integration des entwickelten Tools mit anderen Anwendungen zu fördern. Eine Anbindung an das GIS und die mit der Parzelle verbundenen Baureglements wäre sinnvoll. So könnten Benutzer das Tool einfacher nutzen und die Datenintegration wäre nahtloser. Eine solche Integration könnte auch die automatische Aktualisierung von Daten ermöglichen, was die Genauigkeit und Zuverlässigkeit der durchgeführten Machbarkeitsstudien weiter erhöhen würde. Der Skepsis, ausgehend von der Automation solcher Abläufe, dass das Ergebnis Fehler beinhaltet, könnte so entgegengewirkt werden.

Eine weitere Richtung bezüglich der Weiterführung des Prototypen wäre die Einbindung mehrerer Aspekte wie Energieeffizienz, Ressourcenverbrauch und Rückbaukosten. Diese Parameter könnten es ermöglichen, eine noch umfassendere und detailliertere Machbarkeitsstudie durchzuführen. Die Integration solcher Funktionen könnte die Anwendung des Tools über den ursprünglichen Kontext hinaus erweitern und erheblichen Mehrwert für Benutzer liefern.

Schliesslich wäre es nötig, Langzeitstudien und Fallstudien durchzuführen, bei denen das in dieser Arbeit entwickelte Tool in realen Projekten eingesetzt wird. Mit dem Ziel, Feedback vom Benutzer zu erlangen, um Verbesserungspotenziale zu identifizieren. Das könnte dazu beitragen, die Benutzerfreundlichkeit zu verbessern, die Anpassungsfähigkeit zu erhöhen und die Robustheit des Tools im Hinblick auf unterschiedliche Projektanforderungen der verschiedenen Gemeinden zu testen. Insgesamt bieten die erzielten Ergebnisse dieser Arbeit eine solide Grundlage für weiterführende Forschungen im Bereich digital unterstützter Machbarkeitsstudien. Sie unterstreichen das Potenzial von digitalen Tools, um erhebliche Effizienzsteigerungen zu ermöglichen.

## 8 Reflexion

Im Laufe dieser Arbeit galt es, viele Hürden zu überwinden, um Kenntnisse und Fähigkeiten im Umgang mit digitalen Technologien zu erweitern. Eine der schwierigsten Aufgaben war die Entwicklung des Prototyps, die ein tieferes Verständnis der Dateiformate, Programme und der Möglichkeiten von Python erforderte. Das Betreten von Neuland mit Python und Grasshopper erforderte Geduld, um Lösungen für Probleme zu finden. Es stellte sich heraus, dass die Stolpersteine meist dort lagen, wo keine Probleme erwartet wurden. Trotz der Hindernisse, die während des gesamten Arbeitsprozesses auftraten, war dieses Projekt eine lohnende Erfahrung. Das Verfassen und Verknüpfen von einem Code, der auf jedem Computer läuft und auf verschiedene Parzellen anwendbar ist, hat das gesamtheitliche Verständnis erheblich erweitert. Es ist nun besser abschätzbar, was möglich ist, selbst bei Themen, die über Machbarkeitsstudien hinausgehen.

Dieses Projekt hat einen entscheidenden Aspekt aufgezeigt: Gut strukturierte und organisierte Daten sind für die Nutzung digitaler Technologien von grösster Bedeutung. Bei allen zukünftigen Bemühungen im Baubereich wird es Ziel sein, eine kontinuierliche Verbesserung in der Datenorganisation/Strukturierung zu erreichen. Denn nur aus einer guten Struktur kann eine noch bessere Struktur entstehen. Im Rahmen dieser umfassenden Aufgabe ist es jedoch wichtig zu erkennen, dass dieser Prototyp - zusammen mit den methodischen Grundlagen - nur ein Schritt auf einem langen Weg zu effizienteren Machbarkeitsstudien ist. Wie die Bearbeitung von dem Projekt gezeigt hat, ist es von entscheidender Bedeutung zu erkennen, dass trotz der rasanten Entwicklungen in der digitalen Technologie bestimmte Aspekte nicht einfach automatisiert werden können und sollten. Die Bedeutung der menschlichen Intuition und Kreativität muss auch in einer Welt, in der die Automatisierung vorherrscht, einen hohen Stellenwert haben. Insgesamt stellte die Arbeit an diesem Projekt eine bereichernde Erfahrung dar und es besteht die Aussicht, das erworbene Wissen in neuen Projekten anzuwenden.

## 9 Literaturverzeichnis

Archistar. (n.d.). *About us*. Aufgerufen am 01.03.2023 von <https://www.archistar.ai/about/>

Architektvergleich.ch. (n.d.). *Machbarkeitsstudien - Nicht nur bei Bauprojekten die Grundlage für unternehmerischen Erfolg*. Aufgerufen am 14.05.2023 von <https://www.architektvergleich.ch/ratgeber/machbarkeitsstudien-nicht-nur-bei-bauprojekten-die-grundlage-fuer-unternehmerischen-erfolg-c:400087#:~:text=Machbarkeitsstudien%20werden%20vor%20allem%20bei,oft%20einem%20grossen%20Risiko%20ausgesetzt.>

Auxalia. (n.d.). *Autodesk Spacemaker*. Aufgerufen am 01.03.2023 von <https://www.auxalia.com/software/autodesk-spacemaker/>

Bozikovic, A. (2022). *The End of Sidewalk Labs*. Aufgerufen am 03.03.2023 von <https://www.architecturalrecord.com/articles/15573-the-end-of-sidewalk-labs>

Bundesamt für Statistik. (01.01.2023). *Die 2136 Gemeinden der Schweiz am 1.1.2023*. Aufgerufen am 01.03.2023 von <https://www.bfs.admin.ch/bfs/de/home/statistiken/kataloge-datenbanken.assetdetail.24065856.html>

Czogalla, J. (2010). *Kernaussagen der Diskursethik von Jürgen Habermas*. Aufgerufen am 15.04.2023 von <https://philosophisch-ethische-rezensionen.de/rezension/Themen/Habermas1.html>

Dreier, D. (2022). *Bauland in der Schweiz: Wie viel kostet der Wohntraum?*. Aufgerufen am 26.05.2023 von <https://www.moneyland.ch/de/bodenpreise-bauland-schweiz#:~:text=Die%20Bodenpreise%20f%C3%BCr%20Einfamilienh%C3%A4user%20sind,mitt%20150%20Franken%20pro%20Quadratmeter.>

ETH Zürich. (2018). *Machbarkeit*. Aufgerufen am 18.03.2023 von <https://map.arch.ethz.ch/artikel/2/machbarkeit>

FAB-Solutions. (2023). *Machbarkeitsstudie*. Aufgerufen am 22.05.2023 von <https://www.fabsolutions.ch/machbarkeitsstudie/>

Fischer Architekten AG. (2017). *Dossier Machbarkeitsstudien*. Aufgerufen am 22.03.2023 von [https://www.fischer-architekten.ch/sites/default/files/2018-03/dossier\\_machbarkeitsstudien\\_2017\\_fischer\\_architekten\\_web.pdf](https://www.fischer-architekten.ch/sites/default/files/2018-03/dossier_machbarkeitsstudien_2017_fischer_architekten_web.pdf)

Frausto-Robledo, A. (17.11.2020). *Details on Autodesk's Spacemaker Acquisition*. Aufgerufen am 01.03.2023 von <https://architosh.com/2020/11/details-on-autodesks-spacemaker-acquisition/>

Gemeinde Freienbach. (2022). *Baureglement Gemeinde Freienbach*. Aufgerufen am 01.03.2023 von [https://www.freienbach.ch/\\_docn/1403057/Baureglement\\_Freienbach\\_20181201.pdf](https://www.freienbach.ch/_docn/1403057/Baureglement_Freienbach_20181201.pdf)

Gemeinde Hombrechtikon. (1998). *Bau- und Zonenordnung*. Aufgerufen am 01.03.2023 von [https://www.hombrechtikon.ch/public/upload/assets/458/171016\\_Bau\\_und\\_Zonenordnung.pdf?fp=1](https://www.hombrechtikon.ch/public/upload/assets/458/171016_Bau_und_Zonenordnung.pdf?fp=1)

Gemeinde Lachen. (1995). *Planungs- und Baureglement (PBR)*. Aufgerufen am 01.03.2023 von [https://www.lachen.ch/\\_docn/2607424/R\\_20200317\\_PBR\\_aktuell\\_Version\\_Homepage.pdf](https://www.lachen.ch/_docn/2607424/R_20200317_PBR_aktuell_Version_Homepage.pdf)

Gibson, E. (20.10.2020). *Sidewalk Labs creates machine-learning tool for designing cities*. Aufgerufen am 05.03.2023 von <https://www.dezeen.com/2020/10/20/delve-sidewalk-labs-machine-learning-tool-cities/>

Haustechnik Dialog. (24.08.2016). *Digitalisierung der Baubranche: Wer nicht mithält, riskiert seine Wettbewerbsfähigkeit - Unternehmen erkennen Notwendigkeit, setzen aber zu zögerlich um*. Aufgerufen am 16.03.2023 von <https://www.haustechnikdialog.de/News/Images.aspx?ID=18307&Number=1>

McDonough, W., & Braungart, M. (2002). *Cradle to Cradle: Remaking the Way We Make Things*. Aufgerufen am 07.03.2023 von <https://www.cambridge.org/core/journals/environmental-practice/article/abs/book-review-cradle-to-cradle-remaking-the-way-we-make-things-william-mcdonough-and-michael-braungart-2002-north-point-press-new-york-208-pp-2750-paperback-9780865475878/23195546CBDCBDABABED98E6E3F5CF00>

McDonough, W., & Braungart, M. (22.04.2002). *Cradle to Cradle: Remaking the Way We Make Things*. North Point Press. New York.

McNeel. (2020). *Using Numpy SciPy and other CPython libraries in GHPython*. Aufgerufen am 28.02.2023 von <https://wiki.mcneel.com/grasshopper/cpython>

Python Software Foundation. (2023). *Python 3.10.11 documentation*. Aufgerufen am 29.02.2023 von <https://docs.python.org/3/>

Shamanth Kumar M, E. (2022). *Digitization in Construction Industry*. Aufgerufen am 13.04.2023 von <https://constrofacilitator.com/digitization-in-construction-industry/>