



eBKP⁺

eBKP-H Klassifizierung

Ein pyRevit-Plugin von Nicole und Orlando

Konzept

- Datenauswertung mit **pyRevit** direkt aus dem Revit Modell
- Schnelles, modellbasiertes **Ausmass nach eBKP-H**
mit Anthropic's **Haiku** KI-Modell
- Visualisierung und Export mit **Streamlit**

Entwicklungsumgebung

- Visual Studio Code als IDE
- Virtual Environment mit Python **3.13.7**
- pyRevit als Schnittstelle zwischen Python und Revit
- GitHub Repository

zusätzliche Tools:

- Mermaid Flowchart-Diagramme erstellen
- Marp Slides aus Markdown-Syntax

Python-Bibliotheken

Datenanalyse

```
numpy          # Numerische Berechnungen und Arrays  
pandas        # Tabellen–Verarbeitung und Analyse
```

Visualisierung

```
matplotlib    # Datenplotting und Visualisierung  
plotly         # Diagramme und Dashboards.  
streamlit      # Apps und Dashboards.
```

Sonstige nützliche Tools

```
openpyxl       # Verarbeitung von Excel–Dateien (.xlsx).
```

Herausforderungen

- Datenexport mit Schichtaufbau aus Revit
- Prompt für eBKP-H Kategorisierung > Antwort in **JSON** Format.

Erfolge

- eBKP-H Kategorisierung mit KI
- Export aus pyRevit
- übersichtliche Visualisierung in Streamlit

STREAMLIT

Klassifizierung mit Haiku
Visualisierung der Ergebnisse
Kostenberechnung

KI-BKP-Klassifizierung

localhost:8501

HSLU DC

Deploy :

Einstellungen

Batch-Modus (empfohlen)

Batch-Größe: 40

Debug-Modus

Konfidenz-Schwellenwert: 0.70

Klassifizierungs-Ergebnisse

Klassifizierte Elemente: 17

Ø Konfidenz: 91.2%

Niedrige Konfidenz: 0 ($\uparrow 0.0\%$)

Verschiedene BKP-Codes: 8

Konfidenz-Verteilung

Verteilung der KI-Konfidenz

Konfidenz	count
0.80 - 0.85	2
0.85 - 0.90	1
0.90 - 0.95	5
0.95 - 1.00	8

Top 10 BKP-Codes

Häufigste BKP-Codes

BKP Code	Anzahl
G01	3
E03	3
D01	3
J01	3
G04	2
I01	1
C01	1
G05	1

Qualitätskontrolle

Alle Elemente haben eine Konfidenz über 70%

The screenshot shows a dark-themed user interface for a machine learning classification application. On the left, a sidebar contains 'Einstellungen' (Settings) with a 'Batch-Modus' toggle (on), 'Batch-Größe' slider (40), 'Debug-Modus' toggle (off), and 'Konfidenz-Schwellenwert' slider (0.70). Below this is the 'API Status' section, which shows an 'API Key aktiv' button (green background with white text) and details like 'Key: sk-ant-api...fQAA' and 'Quelle: .env Datei'. The main area is a log history with colored bars indicating success (blue), warning (yellow), and error (red). The logs are as follows:

- 21:02:50 | Klassifizierung gestartet
- 21:02:50 | Modus: Batch
- 21:02:50 | 17 Elemente zu klassifizieren
- 21:02:50 | eBKP-H Classifier initialisiert...
- 21:02:50 | API-Response-Log wird gespeichert: ebkp_classification_log_20251129_210250.txt
- 21:02:50 | Batch-Verarbeitung mit Größe 40
- 21:02:58 | Batch 1/1 abgeschlossen (17 Elemente)
- 21:02:58 | ✓ Klassifizierung erfolgreich abgeschlossen
- 21:02:58 | Durchschnittliche Konfidenz: 91.2%
- 21:02:58 | 17 API-Responses aufgezeichnet
- 21:02:58 | Detailliertes Log gespeichert: ebkp_classification_log_20251129_210250.txt
- 21:02:58 | Fehler: 'url.pathname'

The screenshot shows a dark-themed application window titled "KI-BKP-Klassifizierung" running on "localhost:8501". The left sidebar contains "Einstellungen" (Batch-Modus (empfohlen), Batch-Größe 40, Debug-Modus, Konfidenz-Schwellenwert 0.70) and "API Status" (API Key aktiv ✓, Key: sk-ant-api...fQAA, Quelle: .env Datei). The main area is titled "KI API Responses (Echtzeit)" and displays 17 recorded API requests. Each request card includes "Request Info:", "Parsed Result:", and "Raw API Response".

Request #17 | 21:02:58.213 | Element: 0915 x 1220mm (Fenster)...

Request Info:
Zeitstempel: 21:02:58.213
Request #: 17
Element: 0915 x 1220mm (Fenster)

Parsed Result:
✓ BKP: E03 (Konfidenz: 90.0%)
Beschreibung: Element in Aussenwand

Raw API Response:

```
{"code": "E03", "desc": "Element in Aussenwand", "conf": 0.9}
```

Request #16 | 21:02:58.213 | Element: Interior - 138mm Partition (1-hr) (Waende)...

Request Info:
Zeitstempel: 21:02:58.213
Request #: 16
Element: Interior - 138mm Partition (1-hr) (Waende)

Parsed Result:
✓ BKP: G01 (Konfidenz: 95.0%)
Beschreibung: Trennwand, Innentür, Innenstor

Raw API Response:

```
{"code": "G01", "desc": "Trennwand, Innentür, Innenstor", "conf": 0.95}
```

Erkenntnisse

- ~~Einzelne Bauteil-Kategorien pro Liste Auswerten~~
 - Alle Bauteil-Kategorien in einer Liste Auswerten
- Claude funktioniert viel besser mit `#TODO Änderung` als reinem umschreiben vom Problem das gelöst werden soll

Nächste Schritte

1. Alle Elemente in einer ~~Excel~~ Liste exportieren
2. Elemente mit ~~BKP~~ klassifizieren
3. ~~Streamlit~~ Dashboard erstellen
4. Auswertung als ~~PDF~~ exportieren
5. Kostenberechnung nach verschiedenen Methoden

Optional

1. Flächen-/ Volumenauswertung nach **SIA416**

“

If a picture is worth a thousand words,
a prototype is worth a thousand meetings,

— *Tom & David Kelley, IDEO*

”