



# ohne Dachschaden Preventa GVB

---

Technische Informationen für die Jury



## Technische Informationen für die Jury

### Aktueller Stand des Sourcecodes

- <https://github.com/ooemperor/ohneDachschaden>
- Frontend inkl. Backend funktional.
- Datenbank mit Datenmodell und Dataload vollständig funktional und automatisiert.

### Ausgangslage

- Schnelle Übersicht über mögliche Gefahren an meinem Standort.
- Anzeige der Eintrittswahrscheinlichkeit pro Gefahr und Standort (GVB API)
- Vorschläge für die Gefahrenprävention basieren auf den Gefahren an einem beliebigen Standort.
- Übersicht über Lösungen von anderen Personen in Form von Bildern ersichtlich.
- AI-Anbindung via externen APIs.
- Spezielle Prompts für optimale Lösungsvorschläge zur passenden Gefahr.
- Aufgrund Datengrundlage/Datenmangel keine Entwicklung von eigenem Model.
- Viele Daten vorhanden, keine generelle Übersicht.

### Technischer Aufbau

- Vaadin/Springboot für Frontend inkl. Backend.
- Postgres als Datenbank.
- Python für Webscraping und inserts in Database
- Shell für einfache Web fetches von Daten.
- Docker als Container Engine. (Kubernetes tauglich)

### Implementation

- Ein modularer Aufbau, der es ermöglicht verschiedene Technologien zu benutzen, um die Daten zu laden.
- Einfach zu deployen dank docker compose. Alle Daten inserts sind vollständig in Code automatisiert, was es erlaubt die Ergebnisse zu reproduzieren und periodisch zu aktualisieren.

### Abgrenzung / Offene Punkte

- Keine Implementation von eigenem Machine Learning Model oder AI-Agent aufgrund mangelnder Daten.
- Externe AI Agents für schnelle Vorschläge basierend auf Prompt Engineering.
- Anbindung von weiteren Datenquellen jederzeit möglich.
  - Via API
  - Via AI-Agent (API)
  - Via Import in Datenbank
- Von GVB akzeptierte Bilder nicht implementiert aber angedacht (Approved Badge)