D1: List of Key Terms & Concepts

Actionable Knowledge Representation

Giulio Billi Cono Cirone

November 13, 2025

Project Description

Domain: The project operates in the domain of activity recommendation and tourism within German cities. It focuses on modeling and organizing knowledge about various leisure activities, venues, and events that people can engage in, depending on factors such as city, budget, and weather conditions. The domain aims to capture the relationships between these entities, for example: how an activity is linked to a city, what budget category it belongs to, or whether it is suitable for indoor or outdoor conditions.

Goal: To develop a knowledge-based companion that recommends activities across some German cities based on user preferences. More specifically, the user can choose the: City, Budget (Free, Low, Medium, High), Weather Suitability (Indoor, Outdoor), and Date/Hour. Using these informations, the system will generate meaningful recommendations on what to do based on user preferences.

Key User Questions

The final knowledge graph will be able to answer key questions, such as:

- "What are some free outdoor activities that I can do today in Berlin?"
- "Since today it's raining, which museum could I visit in Munich?"
- "Which nature parks in Stuttgart are open right now?"
- "I'd like to do something outdoors in Hamburg with a medium budget, what do you recommend?"
- "Are there any guided tours in Berlin?"

Conceptual Model Diagram & Notation

The following diagram provides a high-level overview of the conceptual model. The lists in the following sections provide the formal definitions for each component.

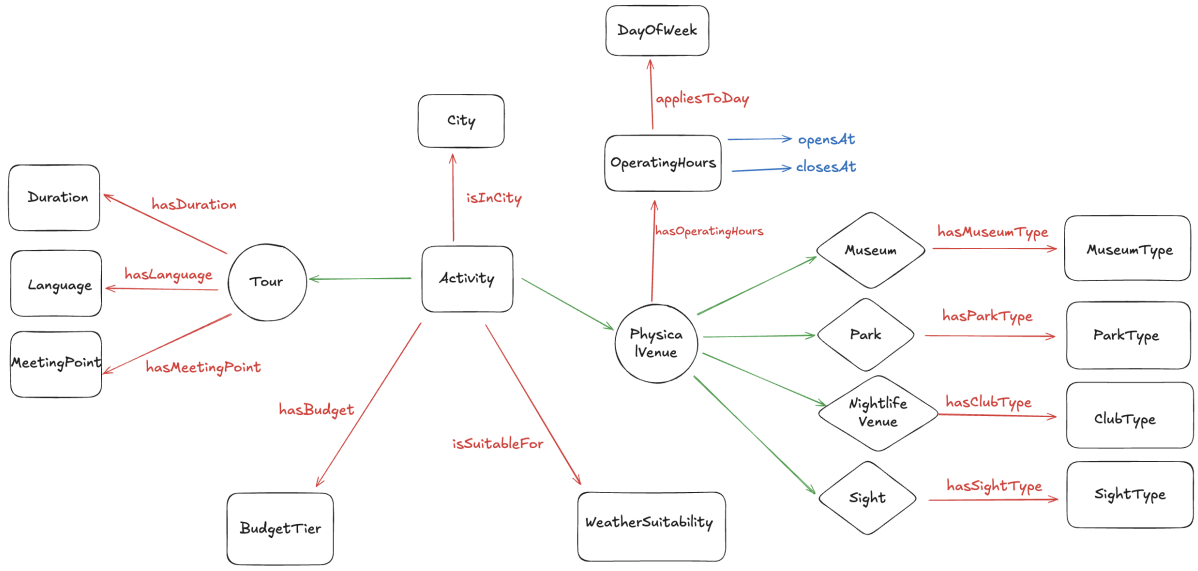

Figure 1: A visual representation of the conceptual model, showing all classes and their relations.

Notation

The diagram can be read using the following notation:

- **Boxes, Circles, & Diamonds:** These shapes represent the Concepts (Classes) of the model (e.g., `Activity`, `Tour`, `Museum`).
- **Green Arrows:** Represent the Subclass relationship. The arrow points from the parent class to the child class (e.g., `PhysicalVenue` → `Museum`).
- **Red Arrows:** Represent Relations (Object Properties), which connect one Class to another Class (e.g., `Activity` → `isInCity` → `City`).
- **Blue Arrows:** Represent Attributes (Data Properties), which connect a Class to a literal value (e.g., `OperatingHours` → `opensAt` → `time`).

1 Key Concepts (Classes)

This list contains the concepts (classes) relevant to modeling the project domain.

Concept (Class)	Definition & Purpose
Activity	A general concept for an activity of interest. Used to represent the common properties of both physical locations and events (e.g., budget, location).
PhysicalVenue	A physical location with set operating hours. Used to represent a subclass of ‘Activity‘ that people can visit, such as museums, parks, or clubs.
Tour	A guided activity to visit the city. Used to represent a subclass of ‘Activity‘ that is an event, not a static location.

Concept (Class)	Definition & Purpose
Museum	An institution displaying artifacts of cultural, artistic, or scientific interest. Used to represent a specific subclass of ‘PhysicalVenue‘.
NightlifeVenue	A location for evening entertainment, such as a bar or club. Used to represent a specific subclass of ‘PhysicalVenue‘ focused on nightlife.
Park	A public green space or natural area. Used to represent a specific subclass of ‘PhysicalVenue‘, such as city parks, zoos, or botanical gardens.
Sight	A location or structure of historical or architectural interest. Used to represent a specific subclass of ‘PhysicalVenue‘, such as landmarks or historical monuments.
City	A German city where an activity is located. Used to model the geographic location of activities within the project's scope (e.g., Berlin, Hamburg, Munich).
BudgetTier	A classification representing the cost of an ‘Activity‘. Used to categorize activities by their price range (e.g., Free, Low, Medium, High).
WeatherSuitability	A classification of an ‘Activity‘'s suitability for different weather conditions. Used to represent whether an ‘Activity‘ is primarily 'Indoor' or 'Outdoor'.
OperatingHours	A concept to model a set of opening/closing times. Used to connect a venue to its hours for a specific day or set of days.
DayOfWeek	A day of the week. Used to represent the specific day(s) an ‘OperatingHours‘ entry applies to (e.g., Monday, Tuesday).
MuseumType	A classification for a museum. Used to categorize museums by their subject (e.g., Art, History, Science).
ParkType	A classification for a park. Used to categorize parks by their type (e.g., Botanical Garden, Zoo, Public Park).
ClubType	A classification for a nightlife venue. Used to categorize venues (e.g., Bar, Club, Disco).
SightType	A classification for a sight. Used to categorize landmarks (e.g., Monument, Architectural Building).
Duration	A length of time. Used to represent how long a tour lasts.
Language	A human language. Used to represent the languages a tour is offered in.
MeetingPoint	A physical location. Used to represent the starting location for a tour.

2 Key Relations (Properties)

This list contains the relations between the classes defined above.

Relation (Property)	Definition & Purpose
isInCity: Activity → City	A relation connecting an activity to the city it is in. Used to model the geographic location of an ‘Activity‘.
hasBudget: Activity → BudgetTier	A relation connecting an activity to its cost classification. Used to model the price range of an ‘Activity‘.
isSuitableFor: Activity → WeatherSuitability	A relation connecting an activity to its weather suitability. Used to model if an ‘Activity‘ is for ’Indoor’ or ’Outdoor’ conditions.
hasOperatingHours: PhysicalVenue → OperatingHours	A relation connecting a physical venue to its set of operating hours. Used to find hours for specific days.
appliesToDay: OperatingHours → DayOfWeek	A relation connecting an operating hours entry to a day of the week. Used to query for hours on a specific day (e.g., "Sunday").
opensAt: OperatingHours → xsd:time	An attribute describing the opening time for an ‘OperatingHours‘ entry. Used to store the opening time (e.g., "09:00").
closesAt: OperatingHours → xsd:time	An attribute describing the closing time for an ‘OperatingHours‘ entry. Used to store the closing time (e.g., "18:00").
hasDuration: Tour → Duration	A relation connecting a tour to its duration. Used to represent how long a tour will take.
hasLanguage: Tour → Language	A relation connecting a tour to its language(s). Used to find tours offered in a specific language.
hasMeetingPoint: Tour → MeetingPoint	A relation connecting a tour to its starting location. Used to inform users where to meet.
hasMuseumType: Museum → MuseumType	A relation connecting a museum to its category. Used to classify and find museums of a specific type.
hasParkType: Park → ParkType	A relation connecting a park to its category. Used to classify and find parks of a specific type.
hasClubType: NightlifeVenue → ClubType	A relation connecting a nightlife venue to its category. Used to classify and find nightlife venues of a specific type.
hasSightType: Sight → SightType	A relation connecting a sight to its category. Used to classify and find landmarks of a specific type.