UNIVERSITY
OF TRENTO - Italy


Dipartimento di Ingegneria e Scienza dell'Informazione

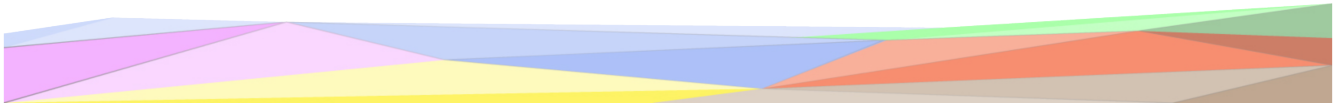
– KnowDive Group –

# KGE 2024 - Student life in Trento

---

| Document Data: | Reference Persons: |
|---|---|
| November 13, 2024 | Davide Cavicchini, Yesun-Erdene Jargalsaikhan |





# Index:


**1 Introduction** **1**

**2 Purpose Definition** **1**

2.1 Informal Purpose . . . 1
2.2 Domain of Interest . . . 1
2.3 Personas & Scenarios definition . . . 2
2.3.1 Personas . . . 2
2.3.2 Scenarios . . . 2
2.4 Competency Questions . . . 3
2.5 Concepts Identification . . . 4
2.6 ER modeling . . . 4

**3 Information Gathering** **7**

3.1 Sources identification . . . 7
3.2 Data sources . . . 8
3.2.1 SmartUnitn2 . . . 8
3.2.2 Trentino Trasporti . . . 8
3.2.3 Open Street Map . . . 8
3.2.4 Points of Interest in Trentino . . . 8
3.3 Datasets cleaning . . . 9
3.3.1 PoI & OSM Data . . . 9
3.3.2 Trentino Trasporti Data . . . 9
3.3.3 SmartUnitn2 Data . . . 10
3.4 Datasets standardization . . . 11
3.4.1 Static Entities . . . 11
3.4.2 Data Streams . . . 13


# Revision History:

| Revision | Date | Author | Description of Changes |
|---|---|---|---|
| 0.1 | October 30, 2024 | Davide Cavicchini, Yesun-Erdene Jargalsaikhan | Completed Phase1 - Purpose defiition |
| 0.1 | November 13, 2024 | Davide Cavicchini, Yesun-Erdene Jargalsaikhan | Completed Phase2 - Information Gathering |

# 1 Introduction

In this project, we will develop a knowledge graph by integrating the following data source. The iLog app, which collects information from students studying at the University of Trento, Open Street Map, the open data resource for geographic data, and Trentino Trasporti for the public transportation data in Trento. The resulting knowledge graph will be utilized to facilitate informed decision-making.

# 2 Purpose Definition

UPDATES

After working with the data at our disposal we had to make several changes to this first phase.
Regarding the competency questions we had to drop the ones asking about the crowdedness stat of a place/bus since the data was not expressive enough to be able to answer them.
While the ER model got a make-over to use a more data and expandability-friendly schema. Now except for the static objects (students, location, and busses), all the other entities have been remodeled as data streams grounded in time and with a duration. In the related section we define how this allows for much better flexibility and extensibility.

This section introduces the purpose, domain of interest, scenarios and personas, competency questions, and concepts identification for the project.

## 2.1 Informal Purpose

The objective of this project is to build a knowledge graph that assists students in planning their trips from one location to another using public transportation in an efficient and comfortable manner. This tool aims to facilitate informed decision-making and enhance students' overall university experience. This will be achieved by integrating historical data on student commutes and activities, public transportation information, and points of interest.

## 2.2 Domain of Interest

To focus our attention to what matters most and what are the distinctive features of the entities we will have to be able to handle.s In this section, we outline in which ways we ground our

representations to the spatial-temporal domain in the world.

As per the spatial domain of the project, we are focusing our attention on the city of Trento, Italy. In particular, we are mostly interested in the commutes and daily activities of students around the city. For these reasons, we identify two main spatial domains of interest:

- Points of interest in Trento that students are interested in, such as bars, libraries
- Public transportation bus stop locations

One interesting domain that we are interested in exploring is the emotional state of students. While this information can be grounded in the world by the location and the time at which it occurs, we also need to define its domain. Following the iLog data format, we use a discrete scale whose values range from 0 to 5.

## 2.3 Personas & Scenarios definition

In this section, we introduce the personas and scenarios to ground the purpose on possible use-case of actual users of our knowledge graph.

### 2.3.1 Personas

To formalize the purpose of the project, we provide personas that covers various lifestyles among student which are useful to define diverse interactions with the knowledge graph.

**Person 1** Alessia, a new international student, has recently started studying at the university.

**Person 2** Paolo, a second-year master's student.

**Person 3** Houda, an Erasmus student who wants to save up money

**Person 4** Lucia, a student habit of dining in restaurants quite frequently

**Person 5** Emanuele, a student who lives in San Bartolomeo student residence

### 2.3.2 Scenarios

For the persons we defined, we described some scenarios students could encounter during their university lifestyle in which our Knowledge Graph can assist for making decisions on planning.

1. **Social Interaction** - Alessia has recently moved to Italy and is excited to spend time with her new friends, exploring the city center of Trento, as she is eager to get to know the city.

2. **University Facilities** - Paolo is a second-year master's student at the University of Trento, currently working on his thesis. He wants to study in a quiet, uncrowded place, so he needs to choose one of the university's facilities.
3. **Daily life** - As an exchange student, Houda has started living in the city center and is planning to go grocery shopping. Since the atmosphere in supermarkets varies, he wants to choose the one that best suits his preferences.
4. **Dinner Place** - Regarding her dining habits, Lucia is looking for decent places to have dinner with her flatmate. While exploring restaurants she had both good and bad experiences with plates, so she doesn't want to choose a bad one.
5. **Personal Activity -** Emanuele is a professional athlete looking to have permanent training at the nearest sports facility to his student residence. A regular commute to the facility is an important part of his daily routine, so he needs to choose the one that will save him time.

### 2.4 Competency Questions

Following the paper on Big-Thick Data generation via reference and personal context unification, what we want to be able to answer are about personal-reference (PR) and reference-personal (RP) context questions. The following is a list of relevant questions for the scenarios and personas we defined which align with the purpose of our Knowledge Graph:

1. **P1-S1**. Is public transport available to reach the destination?
2. **P1-S1**. How many people are currently present on the chosen bus?
3. **P1-S1**. How many people are the social interaction locations in the city?
4. **P2-S2**. Which facility best fits the student's needs or has the least impact on their mood?
5. **P2-S2**. How crowded is BUC?
6. **P3-S3**. Which supermarket best meets the student's needs?
7. **P3-S3**. What was the student's mood when they were at the Coop supermarket?
8. **P3-S3**. What is the best route to the Coop supermarket?
9. **P3-S3**. How did I feel about the trip to the Coop supermarket?
10. **P4-S4**. Which restaurant served a meal that met the student's expectations?
11. **P5-S5**. Which sports facility is closest to the student?
12. **P5-S5**. What is the best bus route to the sports facility?
13. **P5-S5**. What is the closest bus stop to reach the facility?

## 2.5 Concepts Identification

In this section, we try to come up with a mostly complete list and description of what type of concepts we are interested in modeling and which properties are fundamental for the Knowledge Graph. In the next section, we will use this information to guide the modeling of the ER diagram.

| Scenarios | Personas | CQs | Entities | Properties | Focus |
|---|---|---|---|---|---|
| 1-5 | 1-5 | 1-11 | Student | student_id, name, current_position | Contextual |
| 1, 5 | 1, 5 | 1, 13 | Bus Stop | name, direction, time_table, location | Contextual |
| 3, 5 | 2, 5 | 2, 8, 12 | Bus route | number, time_domain, start_time | Contextual |
| 4 | 4 | 10 | Restaurant | civic_number, name, location | Common |
| 2 | 2 | 3-5 | University Facility | civic_number, name, location | Common |
| 5 | 5 | 11 | Sport Facility | civic_number, name, location | Common |
| 1 | 1 | 3 | Bar | civic_number, name, location | Common |
| 5 | 5 | 11 | Residence | intercom_name, civic_number, location | Contextual |
| 3 | 3 | 6-9 | Supermarket | civic_number, name, location | Common |
| 2-4 | 2-4 | 4,7,9-10 | Emotional state | time, duration, location, user, mood | Core |

Table 1: Concepts Identification Table

Table 1 reports the identified concepts and relates each to the specific competency question in which their use is required to correctly answer them. The most important concept we identified for this part is the **Emotional State** which will be used by our Knowledge Graph to address the majority of the reference-to-personal and personal-to-reference queries we aim to tackle.

## 2.6 ER modeling

Having defined the specific concept instances relevant to our identified competency questions, we now aim to formalize the newly acquired insights. To do so, we use the Entity-Relationship (ER) modeling to identify and define the entity types that we will need to manage and use. The resulting model is depicted in the image 1.

The final model we settled on comes from wanting to ground all of the entities and information that will be contained in our KG in the pair `(Student, Time)`. This choice can be easily justified by how our competency questions are formulated and the process we follow to answer each one of them. Initially, it might seem strange to not ground the information also on the location of each event, emotional state, etc... However, we argue that the information about where a particular event or emotional state occurred can be obtained using a simple walk on the KG nodes, by using the path `Stream_1 - Student - Stream_2` and aligning the two using the time coordinate. For example, to answer competency question 7, we can start by collecting all the times Houda went to the Coop supermarket using the location data stream and extracting the time intervals she spent in them. Later I can use this information to intersect all of the Emotional states of Houda which fall into the retrieved intervals we are interested in.

Entity Relationship Diagram

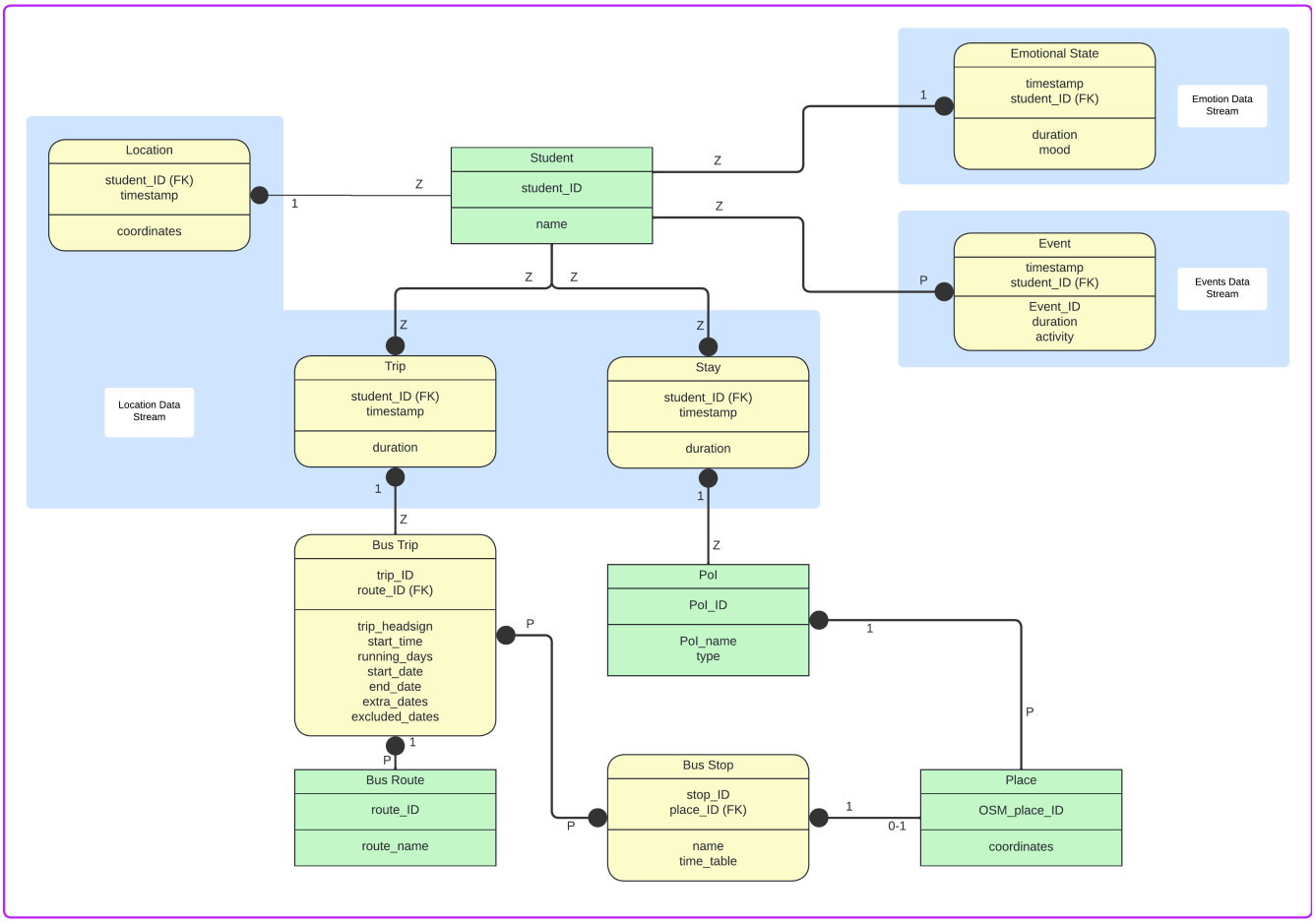



Figure 1: ER model

Additionally, in all the identified questions we are only interested in a subset of places that satisfy specific requirements, such as being suitable for hanging out with friends, having dinner, or working out. Therefore, it was unnecessary to model each place (e.g., restaurant, bar, sports facility) as separate entities. Instead, we collapsed them under a single **PoI** (Point of Interest) entity, differentiated by a *type* property, whose values and definitions we will go over in the next sections.

Using this formulation, we can draw a parallel between how we are storing the information in our entities and the streams of sensor data that might come in from a mobile device. Furthermore, populating the KG with a continuous stream of data from a particular sensor is easy, as we have access to both the `Person` and `Time` coordinate For this project, we are only focusing our attention on modeling the location data stream (GPS), activities, and emotional states, which are more than enough to demonstrate the ability of our KG to answer both PR and RP queries easily. However, it would be relatively trivial to add a new stream of data, as it only needs to be

grounded on the Persons involved and the time interval it takes place in to be able to answer queries about it and jointly retrieve other streams.

- **Location**: Captures the geographical location of a student, identified by `student_ID` and timestamp, which holds coordinates. This represents the continuous tracking of a student's location over time as part of the Location Data Stream.
- **Student**: Represents individual students with a unique `student_ID` and name. It is the central entity connecting various aspects of student activities, movements, and emotional states.
- **Trip**: Documents individual trips taken by students, including `student_ID`, boarding time, duration, and details of boarding and alighting stops. This entity connects to the overall Bus Trip data and represents each instance of a student's travel.
- **Bus Trip**: Defines specific bus journeys, including trip details such as `trip_ID`, route, start and end times, running days, and exceptions in dates. It serves as a template for all the instances of trips students can take.
- **Bus Route**: Represents bus routes with a unique `route_ID` and name. It provides an overarching structure for the bus system in the knowledge graph.
- **Bus Stop**: Details individual bus stops with `stop_ID`, associated place ID, name, and timetable. Each stop connects to both the place it is located in and the route it belongs to.
- **Place**: Describes geographic points of interest, identified by `OSM_place_ID`, with coordinates. These places encompass all relevant points in the student's commute and daily life locations in Trento.
- **PoI (Point of Interest)**: Categorizes points of interest by `PoI_ID`, type, and name, specifying different types of locations like restaurants, libraries, and sports facilities. It helps link the student's stays with specific types of places.
- **Stay**: Represents periods of time a student spends at a location, with `student_ID`, timestamp, and duration. This entity captures instances of visits to locations, making it part of the Location Data Stream.
- **Emotional State**: Tracks the emotional state of students over time, with `timestamp`, `student_ID`, duration, and mood. It provides insights into how students feel in various contexts, forming the Emotion Data Stream.

- **Event**: Records activities that students engage in, with `timestamp`, `student_ID`, event ID, duration, and activity type. This entity is part of the Events Data Stream and captures instances of actions or events students participate in.

# 3 Information Gathering

In this section the second main input for the project is described, namely the data sources. For each resource we report:

- The name, and the description of the information the resource is carrying.
- Type of resource. If it is a language, schema or data value dataset.
- The source from which such resource can be collected.

This section reports the execution of the steps and it is structured in the following subsections:

- Sources identification - where we identify our needs
- Datasets collection - where we identify the data sources we use
- Datasets cleaning - we explain in detail the structure of the data and how we processed it
- Datasets standardization - we show the resulting tables from our processing and the information they contain

## 3.1 Sources identification

In this section, we want to briefly reason on what type of data we need to have access to fulfill the purpose, answer the competency questions, and populate the entities we identified in the previous section. To do so, we have to find and use the following sources of data:

- Trentino Trasporti: we need to have access to information about the bus routes, the stops around the city, and the timetables for each of them.
- PoI in Trento: we need to collect all the places a student from the university of Trento might be interested in, comprehensive of university facilities, bars, restaurants, etc...
- Student Information: we need to have detailed information about the students, comprehensive of their mood state, activities, and positions.

## 3.2 Data sources

We now list and explain the sources we identified and decided to use, which should adequately cover the needs we listed in the previous section. For each dataset, we are interested in explaining what it contains, what data it collects, how to get a copy of such data, and what needs it covers. This will be important to define how we will process this data in the next steps.

### 3.2.1 SmartUnitn2

SmartUnitn2 is a data-value dataset about the everyday life of one hundred fifty-eight university students over four weeks between May and June 2018. The dataset is collected from thirty-four sensors, including location data and responses from questionnaires.
For the purpose of the project, this dataset is used for extracting information about the student commutes and activities including emotional states.
**SmartUnitn2** dataset can be downloaded only through a request form from the authors of the original work, which is ”2018-Smart Unitn 2-Trento”, from this web portal.

### 3.2.2 Trentino Trasporti

Trentino Trasporti is a dataset for public transportation in the Trentino region of Italy. It carries information about public transport stops, lines, schedules, and fare details through the years.
The dataset is crucial for providing bus route, bus stop, and their schedules during the time SmartUnitn2 dataset was collected.
It is available for download via Dati Trentino portal, a website of the public transport agency of the Autonomous Province of Trento.

### 3.2.3 Open Street Map

Open street map is an open-source data-value dataset carries a wide range of geographic data such as geospatial information, map features, metadata for the features, boundaries, and transportation network.
We use this dataset to ground both points of interest and bus stops in a renowned data source, which should facilitate the reuse of the data resulting from this project.
The dataset can be downloaded from Open Street Map (OSM).

### 3.2.4 Points of Interest in Trentino

PoI in Trentino is data-value dataset that provides information on points of interest (PoIs) in the Trentino region of Italy, such as university facilities, bars and restaurants and so on.

We are interested in this dataset to be able to cover all the PoI around Trento that a student might be interested in. The datasets can be downloaded from Open Data Trentino.
However, this data lacks information about university facilities, arguably the most important ones for our project. To get around the limitation we scrape university facility data from OSM using Overpass-Turbo, a web-based data mining tool.

## 3.3 Datasets cleaning

Having identified the sources from which we will gather the data, we now explain its structure and how we processed it to filter out the information we need. To do so, for each dataset we will start by explaining its structure and the data it contains in details. And finally, explain how we extracted the data we need from them.

### 3.3.1 PoI & OSM Data

The Points of Interest in Trentino dataset contains information about individual places in Trentino. We use the OpenStreetMap (OSM) dataset as a ground for localizing both points of interest and student locations. By integrating the PoI dataset with OSM, we match places based on their latitude and longitude coordinates. For our purpose, we download the following source:

- **wu2013poi.json**: `latitude,longitude,social_address,contact,category,name, ..`
  We keep only the name, type, latitude, and longitude of the places in the cleaned dataset.

We scrape the OSM dataset for the points of interest related to university facilities, and obtained following data source:

- **osm_universities.json**: `latitude,longitude,name,amenity/category,building, ...`
  We keep the name, type, latitude, and longitude of the places in the cleaned dataset.

### 3.3.2 Trentino Trasporti Data

The data offered by the Trentino Trasporti agency contains information in the gtfs format for the existing bus routes, the individual trips, the days they run in, the exceptional dates, and the bus stops. For our processing, the files we are interested in are the following:

- **routes.txt**: `route_id,agency_id,route_short_name,route_long_name,...`
  Which we use to get information about the route names and collect the existing `route_id`s
- **stops.txt**: `stop_id,stop_name,stop_lat,stop_lon,...`
  Useful to collect all the bus stops in the city of Trento and their position with which we query the OSM data source to get the `osm_id` for each.

- **trips.txt**: `route_id,service_id,trip_id,trip_headsign,...`
  Which reports all of the bus trips around the city and allows us to connect them to the bus routes using the `route_id` property.
- **stop_times.txt**: `trip_id,arrival_time,departure_time,stop_id,stop_sequence`
  This data allows us to connect all the bus trips to both their path and create the timetables for each one of the bus stops.
- **calendar.txt**: `service_id,monday,tuesday,...,saturday,sunday,start_date,end_date`
  This table is useful to populate the `running_days` property for a particular bus trip by joining it with the trips table over the `service_id` column.
- **calendar_dates.txt**: `service_id,date,exception_type`
  Gives information about the extra dates (`exception_type`$1$) and the excluded ones (2), by merging it with the trip table over the `service_id` column as before.

Note that we focused our attention only on the urban subset of the data, expanding the code to include the extra-urban split of the data should not be expensive.

#### 3.3.3 SmartUnitn2 Data

**3.3.3.1 Time-Diaries (Questionnaires for emotions and activities)** The SmartUnitn2 questionnaire dataset contains information about students' responses to a couple of questions.

For our purpose, we are only interested in responses on what the students were doing, how they were feeling, when they were doing it, and for how long. Therefore, we create separate datasets for student activity history and emotion history. For the emotion data, we represent the emotional character responses as a numerical scale from 0 to 4, and textual definition depending on how down or excited the student was.

**3.3.3.2 GPS Data** The SmartUnitn2 also contains a table with the collected GPS data. This data was collected from the devices with a frequency of one minute, and the related tables have the following relevant columns `userid, latitude, longitude`.

This data is then cross-referenced with the position of the PoIs and the time and position of the busses to construct two tables with all the matches that were found within some margin of error. The data produced uses $50$ meters of radius to catch the near PoIs and bus stops, and a time buffer of $8$ minutes for the bus arrival times.

The resulting data is further processed to filter-out as much noise as possible using some consistency constraints on the routes and stays. In particular, we require the user to have followed a bus trip for more than three stops to be deemed a valid trip and to have stayed at least $10$ minutes in the same place to consider it a stay.

### 3.4 Datasets standardization

In this section, we want to elaborate on the resulting tables and data from this phase. As identified in the previous phase, we have two separate types of information:

**Static Entities** All the entities used to index the information our KG has about the students fall into this category. These correspond to the entities in the ER about buses, locations, and students.

**Data Streams** Collect the information that needs to be indexed using the time. Such streams are used to model the following entities:

- Emotions
- Activities (Events)
- Locations, which is further processed to have the information about the bus trips a student took and their stays at the PoI around Trento.

#### 3.4.1 Static Entities

**3.4.1.1 Students** The data we use for the students is anonymized. For this reason, we do not have access to the names of each student, and the only data we collect to identify them is the unique ids used in the SmartUnitn2 dataset in the **user_ids.csv**.

**3.4.1.2 Locations (PoI)** The data obtained from PoI in Trentino and the scraped data from OSM are merged into poi_and_osm.csv. Here, all the individual places in Trento are provided, including the OSM ID, as well as their identifying name and corresponding category. A sample of the dataset is shown in Figure 2.

| | osm_id | latitude | longitude | name | type |
|---|---|---|---|---|---|
| 0 | 292004245 | 46.076974 | 11.141749 | Biblioteca Argentario | library |
| 1 | 428313995 | 46.051743 | 11.127649 | Biblioteca Clarina | library |
| 2 | 1559090545 | 46.066395 | 11.138857 | BUM - Biblioteca Universitaria Mesiano | library |
| 3 | 1607714595 | 46.006060 | 11.127804 | Biblioteca Mattarello | library |
| 4 | 1721783679 | 46.062734 | 11.124417 | Biblioteca del Polo Umanistico | library |

Figure 2: Point of Interest in Trento

#### 3.4.1.3 Busses

The resulting datasets from Trentino Trasporti are shown in Figures 3, 4, 5 and cover all the information we need to populate our entities and match our users positions to their trips.

| | route_id | route_name |
|---|---|---|
| **0** | 396 | 3 - Cortesano Gardolo P.Dante Villazzano 3 |
| **1** | 400 | 5 - Piazza Dante P.Fiera Povo Oltrecastello |
| **2** | 402 | 7 - Canova Melta Piazza Dante Gocciadoro |
| **3** | 404 | 8 - Centochiavi Piazza Dante Mattarello |
| **4** | 406 | 9 - P.Dante S.Donà Cognola Villamontagna |

Figure 3: Bus routes dataset

In **bus_routes.csv** we collected all the routes served by Trentino Trasporti agency with their unique name, which will be used to give more human-readable responses to the user of our Knowledge Graph.

| | stop_id | stop_name | stop_lat | stop_lon | osm_id | route_id | trip_id | arrival_time |
|---|---|---|---|---|---|---|---|---|
| **0** | 1 | Baselga Del Bondone | 46.078325 | 11.047358 | 1246012567 | [536, 536, 536, 536, 536, 536, 536, 536, 536, ... | ['000289600201709112018060 7', '000289888201709... | ['07:04:00', '07:14:00', '07:46:00', '07:51:00... |
| **1** | 2 | Baselga Del Bondone | 46.078581 | 11.047541 | 1363981295 | [536, 536, 536, 536, 536, 541, 404, 536, 536, ... | ['000289895201709112018060 7', '000289300201709... | ['05:13:00', '06:08:00', '06:20:00', '06:52:00... |
| **2** | 3 | Belvedere | 46.044406 | 11.105342 | 1365224515 | [411, 411, 415, 411, 415, 411, 411, 404, 415, ... | ['000287524201709112018060 7', '000288988201709... | ['06:07:00', '06:28:00', '06:45:00', '06:55:00... |
| **3** | 4 | Lamar Ponte Avisio | 46.134620 | 11.110914 | 307367318 | [543, 543, 543, 543, 543, 543, 543, 543, 543, ... | ['000287126201709112018060 7', '000287127201709... | ['06:27:00', '06:47:00', '06:57:00', '07:04:00... |
| **4** | 5 | Sp 85 Bivio Sopramonte | 46.085226 | 11.069313 | 229059978 | [536, 536, 536, 536, 536, 536, 536, 536, 536, ... | ['000289298201709112018060 7', '000289315201709... | ['05:55:00', '06:25:00', '06:58:00', '07:18:00... |

Figure 4: Bus stops dataset

**bus_stops.csv** contains all the bus `stop_id`s and positions contained in the dataset and merges the data from the bus trips to populate the time tables of each stop. In addition, we also search up the `osm_id` associated to each stop to be able to ground them using the same `Place` entity as the PoIs.

Finally, **bus_trips.csv** collects all the bus routes, merged with the information about the sequence of stops and arrival times, days of the week the route is served ($running\_days$), and the extra and excluded dates.

| | route_id | trip_id | trip_headsign | stops | times | start_time | start_date | end_date | running_days | extra_dates | excluded_dates |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 496 | 00028642120170911201806 07 | Corso Bettini Liceo | [2822, 2823, 2504, 2302, 2301, 2303, 2824, 224... | ['06:50:00', '06:50:00', '06:52:00', '06:54:00... | 06:50:00 | 20170911 | 20180607 | [1, 1, 1, 1, 1, 0, 0] | NaN | [20180602] |
| 1 | 490 | 00028642220170911201806 07 | Viale Dei Colli Sc. Alberghiera | [1311, 1285, 1283, 1373] | ['07:40:00', '07:41:00', '07:43:00', '07:46:00'] | 07:40:00 | 20170911 | 20180607 | [1, 1, 1, 1, 1, 0, 0] | NaN | [20180602] |
| 2 | 490 | 00028642320170911201806 07 | Piazzale Orsi Ist. Veronesi | [1339, 2900, 2343, 2296, 2508, 2478, 1372, 128... | ['08:45:00', '08:46:00', '08:48:00', '08:49:00... | 08:45:00 | 20170911 | 20180607 | [1, 1, 1, 1, 1, 0, 0] | NaN | [20180602] |
| 3 | 572 | 00028642420170911201806 07 | S.Lucia Chiesa | [1284, 1282, 1312, 1431, 1363, 1419, 1418, 275... | ['15:48:00', '15:49:00', '15:50:00', '15:51:00... | 15:48:00 | 20170911 | 20180607 | [1, 1, 1, 1, 1, 0, 0] | NaN | [20180602] |

Figure 5: Bus trips dataset

### 3.4.2 Data Streams

**3.4.2.1 Emotions** The emotional state stream dataset shows the temporal changes in students' moods over time, specifying the duration of each mood. A sample of the dataset is shown in Figure 6

| | userid | timestamp | mood | mood_text | duration |
|---|---|---|---|---|---|
| 0 | 0 | 2018-05-09 19:33:48.395 | 3 | happy | 180.0 min 25.0 sec |
| 1 | 0 | 2018-05-09 22:34:13.508 | 4 | excited | 88.0 min 7.0 sec |
| 3 | 0 | 2018-05-10 12:38:14.671 | 3 | happy | 73.0 min 13.0 sec |
| 4 | 0 | 2018-05-10 13:51:27.778 | 4 | excited | 12.0 min 41.0 sec |
| 5 | 0 | 2018-05-10 14:04:08.978 | 3 | happy | 138.0 min 46.0 sec |

Figure 6: Emotional state dataset

**3.4.2.2 Activities** The activity stream dataset shows the activities of students over time, specifying how long the activity has took place. A sample of the dataset is shown in Figure 7

**3.4.2.3 Locations** The Location stream collects the GPS sensor data of each user. Additionally, it includes higher-level abstractions such as bus trips and stays.

| | userid | timestamp | activity | duration |
|---|---|---|---|---|
| **0** | 0 | 2018-05-09 19:33:48.395 | Guardo Youtube, Serie-Tv, ecc. | 96 min 3 sec |
| **1** | 0 | 2018-05-09 21:09:51.600 | Cinema, Teatro, Concerto, Mostra, ... | 22 min 54 sec |
| **2** | 0 | 2018-05-09 21:32:45.845 | Guardo Youtube, Serie-Tv, ecc. | 120 min 22 sec |
| **3** | 0 | 2018-05-09 23:33:07.865 | Leggo un libro; ascolto musica | 29 min 12 sec |
| **4** | 0 | 2018-05-10 00:02:20.655 | Dormire | 755 min 54 sec |

Figure 7: Activity stream dataset

The **user_poi_stays.csv** file specifies the duration of a student's stay at a location over time.

| | user_id | osm_id | timestamp | duration_minutes |
|---|---|---|---|---|
| **0** | 0 | 2607250269 | 2018-05-17 21:18:55.293 | 12.862083 |
| **1** | 1 | 215037366 | 2018-05-22 23:17:48.593 | 505.479850 |
| **2** | 1 | 275318445 | 2018-05-16 08:24:41.070 | 29.680117 |
| **3** | 1 | 275318445 | 2018-05-16 09:37:21.058 | 44.966133 |
| **4** | 1 | 275318445 | 2018-05-23 08:32:04.545 | 90.657783 |

Figure 8: Student stays in the location dataset

In **user_poi_matches.csv** is collected the data for student locations matching with corresponding PoIs. We also developed a visualization for the extracted data illustrated in Figure 11.

| | userid | osm_id | user_timestamp |
|---|---|---|---|
| **0** | 0 | 2607250269 | 2018-05-13 00:34:53.189 |
| **1** | 0 | 2607250269 | 2018-05-13 21:37:05.588 |
| **2** | 0 | 2607250269 | 2018-05-13 21:37:06.871 |
| **3** | 0 | 2607250269 | 2018-05-13 21:37:07.557 |
| **4** | 0 | 2607250269 | 2018-05-13 21:37:08.979 |

Figure 9: Student location matches with PoI dataset

The result from cross-referencing the students' locations with the bus trips is collected in **user_likely_trips.csv** and it contains information about where each user took the bus, at what time, and where and when they got off. We also developed a visualization for the extracted data which is illustrated in Figure 12.

| | user_id | date | trip_id | boarding_stop_id | alighting_stop_id | boarding_time | alighting_time | duration_minutes | number_of_stops | total_tir |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 2 | 2018-05-18 | 00028695820170911201806 07 | 1318 | 1421 | 2018-05-18 12:36:54.539 | 2018-05-18 12:43:21.746 | 6.453450 | 4 | |
| **1** | 6 | 2018-05-29 | 00028757520170911201806 07 | 407 | 348 | 2018-05-29 12:45:13.865 | 2018-05-29 12:50:45.182 | 5.521950 | 3 | |
| **2** | 6 | 2018-05-30 | 00028926920170911201806 07 | 2426 | 2780 | 2018-05-30 17:22:22.525 | 2018-05-30 17:32:30.575 | 10.134167 | 3 | |
| **3** | 6 | 2018-06-01 | 00028943520170911201806 07 | 349 | 406 | 2018-06-01 17:01:00.655 | 2018-06-01 17:11:59.210 | 10.975917 | 3 | |
| **4** | 6 | 2018-06-04 | 00028919920170911201806 07 | 172 | 348 | 2018-06-04 09:43:12.040 | 2018-06-04 09:57:49.920 | 14.631333 | 4 | |

Figure 10: Student likely trips dataset


Figure 11: Visualization of the extracted stays for a day

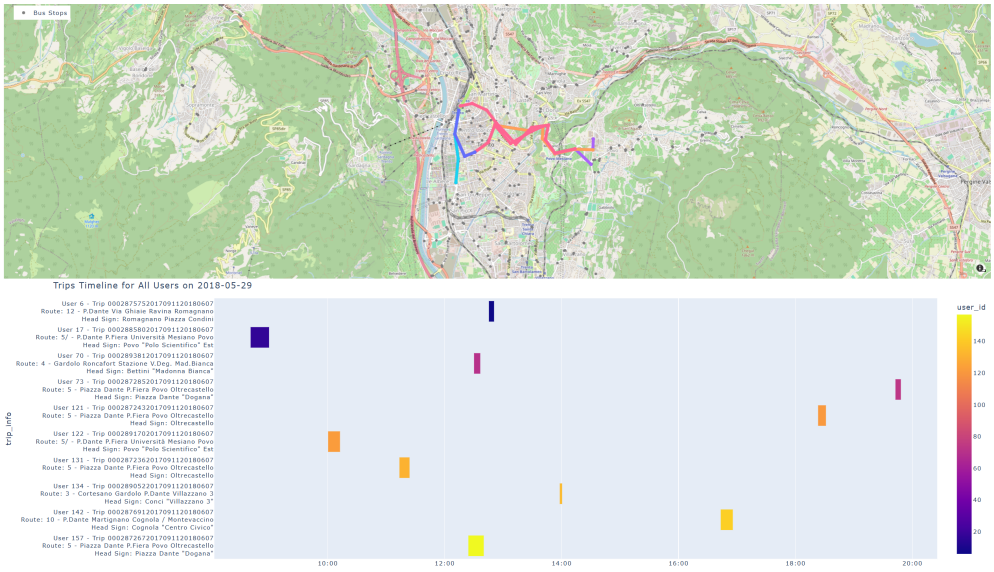



Figure 12: Visualizaton of the extracted trips for a day