

3050 Webprogrammierung und interaktive Datenvisualisierung

Projektarbeit (HS 25)

Zielsetzung

Die Projektarbeit dient der praktischen Wiederholung und Vertiefung der im Modul vermittelten Inhalte und Kompetenzen. Sie bereitet auf die mündliche Prüfung vor und hilft, etwaige Wissenslücken rechtzeitig zu erkennen. Zudem simuliert sie die in der Applikationsentwicklung notwendigen Schritte und kann als Vorbereitung auf spätere Arbeiten (Vertiefungsprojekte, Bachelorarbeit) angesehen werden.

Rahmenbedingungen

Die [Aufgabenstellung](#) wird von euch in **Zweiergruppen** (Ausnahmefall: Dreiergruppe) bearbeitet, wobei ihr die Organisation und Umsetzung der Arbeiten frei gestalten könnt. Die Benutzung von **generativer KI** ist erlaubt.

Das Projekt muss bis zum **21.12.2025** auf Moodle unter Angabe des/der Github-Link(s) eingereicht werden. Die Projektarbeit bildet die Grundlage für die mündliche Prüfung (s.u.), wird jedoch selbst **nicht benotet**.

Hinweis: Ihr werden in der mündlichen Prüfung voneinander separat zu den Inhalten des Moduls sowie eurer Projektarbeit befragt. Dabei wird erwartet, dass ihr Fragen nicht nur zu eurem Beitrag, sondern auch dem eurer Projektpartnerin / Projektpartner sowie dem der KI beantworten könnt.

Datenbasis

In einem Pilotprojekt erhebt die Stadt Zürich zusammen mit Projektpartnern seit 2021 Passantenfrequenzen an vier Standorten entlang der Bahnhofsstrasse Zürich. Mithilfe von Laserscannern wird dabei die Anzahl der Personen pro Stunde, ihre Laufrichtung und Zugehörigkeit (Erwachsene / Kinder) bestimmt. Erfasst werden zudem Wetterverhältnisse und Lufttemperatur.

Die Daten stammen aus dem [Open Data Portal der Stadt Zürich](#). In Moodle findet ihr zwei Versionen:

- **Teildatensatz.json:** Eine ausgedünnte Version des Gesamtdatensatzes. Diese könnt ihr z.B. als vorläufige Datenbasis für die **Frontend**-Entwicklung und Visualisierungen benutzen, bevor ihr den Server und die Kommunikation implementiert.
- **Gesamtdatensatz.csv:** Der Gesamtdatensatz (ca. 26 MB) für die Verwendung im **Backend**. Die finale Version eurer Applikation soll diesen, vollständigen, Datensatz verwenden.

Aufgabenstellung

Entwickelt auf Basis der Modulinhalte und den [Minimalanforderungen](#) eine Web-Applikation (Frontend und Backend), welche zuerst eine der **Fokusfragen** (s.u.) beantwortet und dann die Datenbasis für die visuelle Erkundung durch Nutzende zugänglich macht.

Wählt zunächst eine für euch passende Fragestellung aus, welche die Applikation mit Hilfe einer oder mehrerer geeigneter Visualisierungen beantwortet werden soll. Es ist nachstehend das Fragemuster, sowie Beispielfragestellungen aufgeführt. Ihr könnt eine der Beispielfragen wählen oder anhand des Musters und des Datensatzes eure eigene Fokusfragestellung definieren.

Fragemuster: Differenz (Anteil) von Fussgängerzählungen während eines Zeitraums an einem Ort oder Differenz (Anteil) von Fussgängerzählungen an verschiedenen Orten zu einem (allenfalls aggregierten) Zeitpunkt.

Beispiel Fokusfragen (nach dem Fragemuster):

- Wann (Zeitraum) gibt es an der Bahnhofstrasse Nord (Ort) mehr erwachsene Fussgänger, die in Richtung "ltr" gehen im Vergleich zu Richtung "rtl" (Anteil/Zählung)?
- Wann (Zeitraum) ist der Anteil der Kinder im Vergleich zu den erwachsenen Fussgängern (Anteil/Zählung) am grössten an der Bahnhofstrasse Mitte (Ort)?
- Wo (Raum/mehrere Orte) ist der Anteil der Erwachsenen, die in Richtung ltr gehen (Anteil/Zählung), an einem definierten Tag im April 2024 (Zeitpunkt) am grössten?
- Wo (Raum/mehrere Orte) sind im gesamten April 2024 (aggregierter Zeitpunkt) anteilmässig (d.h. im Vergleich zu allen Passanten) die meisten Kinder unterwegs (Anteil/Zählung)?

In einem zweiten oder ergänzenden Teil soll die Applikation den Nutzenden die **Exploration des Gesamt-Datensatzes** ermöglichen, z.B. um Vergleiche anzustellen oder weitergehende Fragen klären zu können.

Überlegt in der **Planung** eures Vorgehens, welche Funktionalität, Interaktionen und Visualisierungen jeweils nötig sind, wie die Abläufe logisch und effizient angeordnet werden können und wie Informationen, Bedienbarkeit aber auch Feedback den Nutzern kommuniziert werden sollen (siehe Lektion «User-Interface-Design»).

Benutzt in der **Umsetzung** die im Modul vorgestellten Technologien – React, MUI und CSS für das User-Interface, Altair, Vega für interaktive Visualisierungen, React-Leaflet für Webkarten sowie FastAPI für das Backend. Statt einer Datenbank soll der Gesamtdatensatz beim Starten des Backends via in Pandas-DataFrame in den Arbeitsspeicher geladen werden. Beachtet, dass ihr aufgrund der Datenmenge nicht den gesamten Datensatz an das Frontend schicken solltet und somit eine Interaktion zwischen Backend und Frontend benötigt, bei der das Frontend so wenig Daten wie möglich und so viel wie nötig erhält.

Minimalanforderungen

Applikation:

- Die Web-Applikation beantwortet eine der Fokusfragen.
- Die Web-Applikation ermöglicht zusätzlich die Exploration des gesamten Datensatzes: Die Nutzerin kann mithilfe von Interaktionen und/oder Visualisierungen die Datengrundlage nach zeitlichen, räumlichen oder thematischen Aspekten erkunden. Dabei ist darauf zu achten, dass die Applikation performant läuft und nicht zu viele Daten auf einmal verarbeitet werden.
- Es werden die im Unterricht behandelten Technologien verwendet.

Frontend / Interface:

- Das Interface bietet den Nutzenden mehrere unterschiedliche Interaktionsmöglichkeiten (z.B. Navigation, Auswahl, Filter, Darstellungsänderungen) an.
- Das Interface enthält interaktive Visualisierungen und, falls sinnvoll, eine oder mehrere Webkarten.
- Das Interface kann mit dem Server kommunizieren, um Daten oder Visualisierungen anzufragen.

Backend / Server:

- Der Server hält den gesamten Datensatz (Gesamtdatensatz.csv) vor. Teil- oder aggregierte Daten werden vom Frontend über eine Fast-API Schnittstelle und geeignete Endpunkte bzw. Parameter abgerufen.

Finalisierung:

- Die Applikation ist in einer README.md-Datei auf GitHub beschrieben, sodass auch Aussenstehende (z.B. Dozierende) den Inhalt und Fokus des Projekts verstehen.
- Mithilfe der in der README.md angegebenen Informationen (Anweisungen, Befehle) können Aussenstehende (z.B. Dozierende) Frontend und Backend als Prozess(e) auf "localhost" starten und die App(s) testen.