Projektová dokumentace

Implementace systému pro získání, zpracování a ukládání dat

COVID-19

Autoři:
Tomáš Beránek (xberan46)
Daniel Kamenický (xkamen21)
Richard Klem (xklemr00)

Datum odevzdání: **3. listopadu 2021**

Obsah

1 Řešení 1. části projektu

Cílem 1. části projektu je:

- analýza zdrojů dat a jejich dílčích datových sad, zejména struktura dat, atributy, datové typy atributů, možnosti propojení datových sad atp.,
- vytvoření systému pro stažení, předzpracování a vložení dat do MongoDB.

1.1 Volba dotazů ze zadání

Pro vhodné předzpracování dat je dobré znát způsob, jakým se budou data využívat. Na základě této informace je pak možné například – identifikovat nepotřebná data, která lze v předzpracování odstranit, zvolit vhodný databázový systém, upravit formát atributů v předzpracování atp. Cílem 2. části projektu je realizace dvou dotazů typu A, jednoho dotazu typu B, dvou vlastních dotazů a příprava dat pro jednu z dolovacích úloh typu C.

Vybrané dotazy typu A:

1) Vytvořte čárový (spojnicový) graf zobrazující vývoj covidové situace po měsících pomocí následujících hodnot: počet nově nakažených za měsíc, počet nově vyléčených za měsíc, počet nově hospitalizovaných osob za měsíc, počet provedených testů za měsíc. Pokud nebude výsledný graf dobře čitelný, zvažte logaritmické měřítko, nebo rozdělte hodnoty do více grafů.

   Potřebná data:

   – **počet nově nakažených** (minimální granularita – měsíc)

   – **počet nově vyléčených** (minimální granularita – měsíc)

   – **počet nově hospitalizovaných** (minimální granularita – měsíc)

   – **počet nově provedených testů** (minimální granularita – měsíc)

3) Vytvořte sérii sloupcových grafů, které zobrazí:

   – počty provedených očkování v jednotlivých krajích (celkový počet od začátku očkování),

   – počty provedených očkování jako v předchozím bodě navíc rozdělené podle pohlaví. Diagram může mít např. dvě části pro jednotlivá pohlaví,

– počty provedených očkování, ještě dále rozdělené dle věkové skupiny. Pro potřeby tohoto diagramu postačí 3 věkové skupiny (0-24 let, 25-59 let, nad 59).

Potřebná data:

– **počet provedených očkovaní u můžu/žen v daném kraji** (minimální granularita – celkem)

– **počet provedených očkovaní podle věku v daném kraji** (minimální granularita počtu očkování – celkem, minimální granularita věku – 0-24 let, 25-59 let a nad 59 let)

Vybraný dotaz typu B:

2) Vytvořte sérii sloupcových grafů (alespoň 3), které porovnají vývoj různých covidových ukazatelů vámi zvoleného kraje se zbytkem republiky. Jako covidové ukazatele můžete použít: počet nakažených osob, počet hospitalizovaných osob, počet zemřelých, počet očkovaných. Všechny hodnoty uvažujte přepočtené na jednoho obyvatele kraje/republiky. Zobrazte alespoň 12 po sobě jdoucích hodnot (např. hodnoty za poslední rok po měsících).

Potřebná data:

- **celkový počet nakažených podle kraje** (minimální granularita – měsíc)
- **celkový počet zemřelých podle kraje** (minimální granularita – měsíc)
- **celkový počet očkovaných podle kraje** (minimální granularita – měsíc)

Vlastní dotazy zahrnující data alespoň ze dvou zdrojů:

1) Vytvořte sloupcový graf rozdělený podle věku (po 5letých intervalech) obyvatelstva ve vybraném okresu. Každý sloupec vyjadřuje kolik procent obyvatelstva z dané věkové skupiny dostalo COVID-19 za poslední rok (neuvažujte duplicity).

Potřebná data:

– **počet nakažených podle věku a okresu** (minimální granularita věku – 5leté intervaly)

– **počet obyvatel podle věku a okresu** (minimální granularita věku – 5leté intervaly)

2) Vytvořte graf, který ukazuje aktuální počet nakažených mužů a žen od začátku pandemie COVID-19.

   Potřebná data:

   – **počet nakažených podle pohlaví** (minimální granularita – měsíc)

Vybraná dolovací úloha typu C:

1) Hledání skupin podobných okresů z hlediska vývoje covidu a věkového složení obyvatel.

   – Atributy: počet nakažených za poslední 4 čtvrtletí, počet očkovaných za poslední 4 čtvrtletí, počet obyvatel ve věkové skupině 0..14 let, počet obyvatel ve věkové skupině 15 - 59, počet obyvatel nad 59 let.

   – Pro potřeby projektu vyberte libovolně 50 okresů, pro které najdete potřebné hodnoty (můžete např. využít nějaký žebříček 50 nejlidnatějších okresů v ČR).

   Potřebná data:

   – **počet nakažených podle věku a okresů** (minimální granularita věku – 0-24 let, 25-59 let a nad 59 let, minimální granularita datumu – 3 měsíce)

   – **počet očkovaných podle věku a okresů** (minimální granularita věku – 0-24 let, 25-59 let a nad 59 let, minimální granularita datumu – 3 měsíce)

   – **počty obyvatel v jednotlivých okresech podle věku** (minimální granularita věku – 0-24 let, 25-59 let a nad 59 let)

1.2 Analýza datových sad

Součástí 1. části projektu je i analýza datových sad. Zdroje dat, které je možno využít jsou definovány zadáním. Každý zdroj poskytuje řadu dílčích datových sad. Následující seznam obsahuje pouze datové sady, které poskytují potřebná data uvedená v kapitole 1.1. Pro každou dílčí datovou sadu jsou uvedeny informace:

- **URL** – ze které je možno datovou sadu stáhnout,
- **formát** – v jakém formátu jsou data uložena např. CSV, JSON, HTML atp.,
- **struktura** – uvedeno tabulkou.

1.2.1 COVID-19 v ČR: Otevřené datové sady

Zdroj je dostupný zde. Celkem poskytuje 15 dílčích datových sad. V tomto řešení projektu preferujeme formát CSV, proto jsou jako odkazy na dílčí datové sady uvedeny odkazy přímo na data ve formátu CSV. Důvod preferování výběru CSV formátu je uveden v kapitole 1.3.

Přehled osob s prokázanou nákazou dle hlášení krajských hygienických stanic (v2)

Potřebné pro dotaz A1, B2. C1, vlastní dotaz 1 a vlastní dotaz 2.

- **formát**: CSV, JSON.
- **dostupné zde**
- **struktura**:

Jméno	Datový typ	Užitečné
datum	date	ano
vek	integer	ano
pohlavi	string	ano
kraj_nuts_kod	string	ano
okres_lau_kod	string	ano
nakaza_v_zahranici	boolean	ne
nakaza_zeme_csu_kod	string	ne

Přehled vyléčených dle hlášení krajských hygienických stanic

Potřebné pro dotaz A1.

- **formát**: CSV, JSON.
- **dostupné zde**
- **struktura**:

Jméno	Datový typ	Užitečné
datum	date	ano
vek	integer	ne
pohlavi	string	ne
kraj_nuts_kod	string	ne
okres_lau_kod	string	ne

Přehled hospitalizací

Potřebné pro dotaz A1.

- **formát**: CSV, JSON.
- **dostupné zde**
- **struktura**:

Jméno	Datový typ	Užitečné
datum	date	ano
pacient_prvni_zaznam	integer	ne
kum_pacient_prvni_zaznam	integer	ne
pocet_hosp	integer	ano
stav_bez_priznaku	integer	ne
stav_lehky	integer	ne
stav_stredni	integer	ne
stav_tezky	integer	ne
jip	integer	ne
kyslik	integer	ne
hfno	integer	ne
upv	integer	ne
ecmo	integer	ne
tezky_upv_ecmo	integer	ne
umrti	integer	ne
kum_umrti	integer	ne

Celkový (kumulativní) počet provedených testů podle krajů a okresů ČR

Potřebné pro dotaz A1.

- **formát**: CSV, JSON.
- **dostupné zde**
- **struktura**:

Jméno	Datový typ	Užitečné
datum	date	ano
kraj_nuts_kod	string	ano
okres_lau_kod	string	ne
prirustkovy_pocet_testu_okres	integer	ne
kumulativni_pocet_testu_okres	integer	ne
prirustkovy_pocet_testu_kraj	integer	ano
kumulativni_pocet_testu_kraj	integer	ne
prirustkovy_pocet_prvnich_testu_okres	integer	ne
kumulativni_pocet_prvnich_testu_okres	integer	ne
prirustkovy_pocet_prvnich_testu_kraj	integer	ne
kumulativni_pocet_prvnich_testu_kraj	integer	ne

Přehled vykázaných očkování podle profesí (očkovací místo, bydliště očkovaného)

Potřebné pro dotaz A3, B2 a C1.

- **formát**: CSV, JSON.
- **dostupné zde**
- **struktura**:

Jméno	Datový typ	Užitečné
datum	date	ano
vakcina	string	ano
kraj_nuts_kod	string	ano
kraj_nazev	string	ano
zarizeni_kod	string	ne
zarizeni_nazev	string	ne
poradi_davky	integer	ano
indikace_zdravotnik	boolean	ne
indikace_socialni_sluzby	boolean	ne
indikace_ostatni	boolean	ne
indikace_pedagog	boolean	ne
indikace_skolstvi_ostatni	boolean	ne
indikace_bezpecnostni_infrastruktura	boolean	ne
indikace_chronicke_onemocneni	boolean	ne
vekova_skupina	string	ano
orp_bydliste	string	ne
orp_bydliste_kod	integer	ano
prioritni_skupina_kod	integer	ne
pohlavi	string	ano
zrizovatel_kod	integer	ne
zrizovatel_nazev	string	ne
vakcina_kod	string	ne
ukoncujici_davka	boolean	ne

Přehled úmrtí dle hlášení krajských hygienických stanic

Potřebné pro dotaz B2.

- **formát**: CSV, JSON.
- **dostupné zde**
- **struktura**:

Jméno	Datový typ	Užitečné
datum	date	ano
vek	string	ne
pohlavi	string	ne
kraj_nuts_kod	string	ano
okres_lau_kod	string	ne

Epidemiologická charakteristika obcí

Potřebné pro dotaz C1 a vlastní dotaz 1.

- **formát**: CSV, JSON.
- **dostupné zde**
- **struktura**:

Jméno	Datový typ	Užitečné
den	string	ne
datum	date	ne
kraj_nuts_kod	string	ano
kraj_nazev	string	ano
okres_lau_kod	string	ano
okres_nazev	string	ano
orp_kod	integer	ano
orp_nazev	string	ne
obec_kod	integer	ne
obec_nazev	string	ne
nove_pripady	integer	ne
aktivni_pripady	integer	ne
nove_pripady_65	integer	ne
nove_pripady_7_dni	integer	ne
nove_pripady_14_dni	integer	ne

1.2.2 Český statistický úřad

Zdroj je dostupný zde. Poskytuje pouze 1 dílčí sadu dat.

Obyvatelstvo podle pětiletých věkových skupin a pohlaví v krajích a okresech

Potřebné pro dotaz C1 a vlastní dotaz 1.

- **formát**: CSV.
- **dostupné zde**
- **struktura**:

Jméno	Datový typ	Užitečné
idhod	string	ne
hodnota	integer	ano
stapro_kod	string	ne
pohlavi_cis	string	ne
pohlavi_kod	string	ne
vek_cis	string	ne
vek_kod	string	ne
vuzemi_cis	string	ne
vuzemi_kod	string	ne
casref_do	date	ne
pohlavi_txt	string	ne
vek_txt	string	ano
vuzemi_txt	string	ano

1.3 Předzpracování dat

Veškeré datové sady, které jsou potřeba pro řešení dotazů v kapitole 1.1, jsou k dispozici ve formátu CSV. Tento formát je také paměťově úspornější než formát JSON. Z těchto důvodů jsou veškerá data stahována v CSV a před vložení do databáze transformována na JSON, protože MongoDB přijímá pouze JSON respektive BSON. Při samotné transformaci jsou současně odstraňována nepotřebná data (data, která mají ve sloupci "Užitečné"napsáno "ne").

Stažení, transformaci i odstranění dat zajišťuje python modul `src/data_handler.py`. Pro samotnou transformaci je využito knihovny `Pandas`. Výhodou této knihovny je rychlost transformace, nevýhodou vysoká spotřeba RAM u delších JSON souborů (až 10 GB). Alternativou by bylo využití knihoven `csv` a `json` a iterování přes řádky csv souboru a postupné přidávaní do výsledného json souboru. Tento způsob téměř nezatěžuje RAM, ale je mnohonásobně pomalejší. Protože je však možné využít `swapfile` (na OS Linuxového typu), tak byl zvolen přístup s využitím knihovny `Pandas`.

1.4 Vložení dat do databáze

K připojení do MongoDB databáze je použita třída `MongoClient` z knihovny `pymongo`. Je nutné korektně nastavit připojovací řetězec, a to nejlépe konfigurací lokálního souboru `mongo_secrets.py`, který lze vytvořit kopií ze stejnojmenného souboru určeného k veřejné distribuci, konkrétně pojmenovaného `mongo_secrets.py.dist`. Uživatel má v souboru `loader.py` k dispozici nastavení pro připojení na cloudové řešení, anebo nastavení pro připojené na lokální databázový server. Samotné

vložení již předzpracovaných dat do databáze je jednoduché a tento úkon obstarává funkce `load_data` ze souboru `loader.py`. Datové soubory respektive datové sady jsou vkládány postupně, vždy jedna po druhé. V tuto chvíli nejsou volány žádné další funkce (například dotazy přímo nad daty v samotné databázi). Je vytvořen implicitní index na atribut `_id`, jiné indexy nejsou v této fázi projektu použity.

1.5 Návod na spuštění

Prerekvizity, specifické vlastnosti chování, potřebná nastavení a návod na spuštění jsou popsány v dokumentaci v souboru `README.md`, který je umístěn v kořenovém adresáři projektu.