

Big picture - Inžinierske dielo

Dokumentácia k tímovému projektu

Tímový projekt

Tím č. 21

Vedúci: Ing. Ivan Srba, PhD.

Členovia tímu:

Matej Groma
Matej Horváth
Peter Jurkáček
Jozef Kamenský
Adam Kňaze
Kristína Macková
Lenka Pejchalová
Jakub Sedlár

tim21.2018.fiit@gmail.com

Akademický rok: 2018/2019

Posledná zmena: 9. mája 2019

Obsah

1 Úvod	1
2 Globálne ciele	1
2.1 Globálne ciele za zimný semester	1
2.2 Globálne ciele za letný semester	1
3 Celkový pohľad na systém	2
3.1 Slovník pojmov	2
3.2 Opis systému	3
3.3 Model údajov	3
3.4 Reprezentácia pomocou WBS diagramu	5
4 Architektúra nasadenej serverovej časti	7
A Zoznam príloh	9

1 Úvod

Dokument opisuje stav projektu SmartMobility tímu TrafficWatch po ukončení letného semestra, celkovo sme mali 11 šprintov.

V prvej časti opisujeme globálne ciele projektu, následne zobrazujeme celkový pohľad na systém - predstavujeme slovník, uvádzame "big picture"koncept projektu a stručne opisujeme, z akých častí sa skladá.

2 Globálne ciele

2.1 Globálne ciele za zimný semester

Počas zimného semestra sme sa pustili do implementácie systému horizontálne. Kládli sme dôraz na kvalitnú analýzu technológií a stratégií, predtým než sme začali písať akýkoľvek kód. Následne, experimentálne aspekty projektu sa prototypovali, zatiaľ čo druhá časť tímu pracovala na rozbehaní infraštruktúry projektu. Plánom na zimný semester bolo vytvoriť MVP pre náš projekt.

Počas zimného semestra sme dosiahli zrealizovať nasledujúcu funkcionality:

- Detekcia áut, sledovanie prejazdov cez vyznačené zóny
- Mať prepojený systém s fungujúcou infraštruktúrou
- UI na konfiguráciu kamier a zón
- Vizualizovať štatistiky o prejazdoch

2.2 Globálne ciele za letný semester

Na letný semester máme naplánované rozširovať načatú funkcionality projektu.

Do konca letného semestra by sme chceli implementovať:

- Klasifikáciu a re-identifikáciu objektov na kamere
- Spracovanie pohybu objektov
- Konfigurácia kamier na videu
- Spracovávanie štatistík
- Vizualizovanie štatistík na grafoch, na mape a na obrázku alebo live videu

Z plánovaných cieľov na letný semester sa nám podarilo implementovať nasledujúce:

- Spracovanie pohybu objektov - o všetkých sledovaných objektoch si uchováme informácie o ich polohe a rýchlosti, ktoré neskôr zobrazujeme na frontende.
- Konfigurácia kamier na videu - parametre detekcie kamery sa dajú konfigurovať na diaľku, pomocou live videa sa dá sledovať efekt zmeny .
- Spracovávanie štatistík - informácie o pohybe a prejazdoch objektov sa okamžite po prijatí ukladajú a agregujú, aby sme vyhli opakovaným, výpočtovo náročným dopytom pri ich zobrazovaní.
- Vizualizovanie štatistík na grafoch, na mape a na obrázku - informácie o pohybe objektov zobrazujeme 3 rôznymi spôsobmi, každý poskytuje trochu iný pohľad a vzájomne sa dopĺňajú.
- Klasifikáciu a re-identifikáciu objektov na kamere - pre klasifikáciu objektov sme vytvorili fungujúci prototyp, re-identifikáciu sme zatiaľ neriešili.

3 Celkový pohľad na systém

3.1 Slovník pojmov

Slovník pojmov obsahuje všetky pojmy, ktoré sme si v tíme spoločne zdefinovali a toto je kontext v ktorom sa používajú vo všetkých dokumentoch.

Slovenský pojem	Anglický pojem	Opis/definícia
Kamera	Camera	Fyzické zariadenie, na ktorom prebieha záznam a spracovanie videa
Objekt	Object	Entita reálneho sveta
Sledovaný objekt	Detected object	Objekt, ktorý je sledovaný kamerou
Prejazd	Transit	Predstavuje vektor prechodov sledovaného objektu cez zóny
Sledovaná/Monitorovaná oblasť	Monitored Area	Oblasť sledovaná kamerou
Zóna	Zone	Zelený mnohoúhelník definovaný pomocou súradníc vrcholov/pixelov/bodov (x,y), používaný pri sledovaní prechodu
Prechod	Pass	Vstup sledovaného objektu do zóny a výstup zo zóny
Udalosť	Event	Dátová štruktúra posiadaná z kamery na server
Parametre kamery pre detekciu objektu	Camera object detection parameters	Všetky konfigurovateľné hodnoty použité pri detekcií objektov kamerou (zóny, veľkosť sledovaného objektu)
Trajektória sledovaného objektu	Detected object trajectory	Množina bodov, po ktorých sa pohyboval sledovaný objekt
Rýchlosť sledovaného objektu	Detected object velocity	

Klasifikácia	Classification	Rozpoznanie sledovaného objektu (Auto, Dodávka, Kamión, cyklista, chodec, elektricka)
CI (priebežná integrácia)	Continuous Integration	Súbor procesov a metód zabezpečujúcich priebežné testovanie vytváraného kódu, najmä pri pull requestoch
Nasadzovanie	Deployment	Proces automatizovaného nastavenia prostredia, nakonfigurovania a nainštalovania aplikácií na cieľový server

3.2 Opis systému

Obrázok 1 predstavuje zjednodušený pohľad na celkovú architektúru a rozloženie systému. Celkovo možno systém rozdeliť na 3 hlavné časti:

1. Inteligentná kamera
2. Server a úložisko dát
3. Webové rozhranie

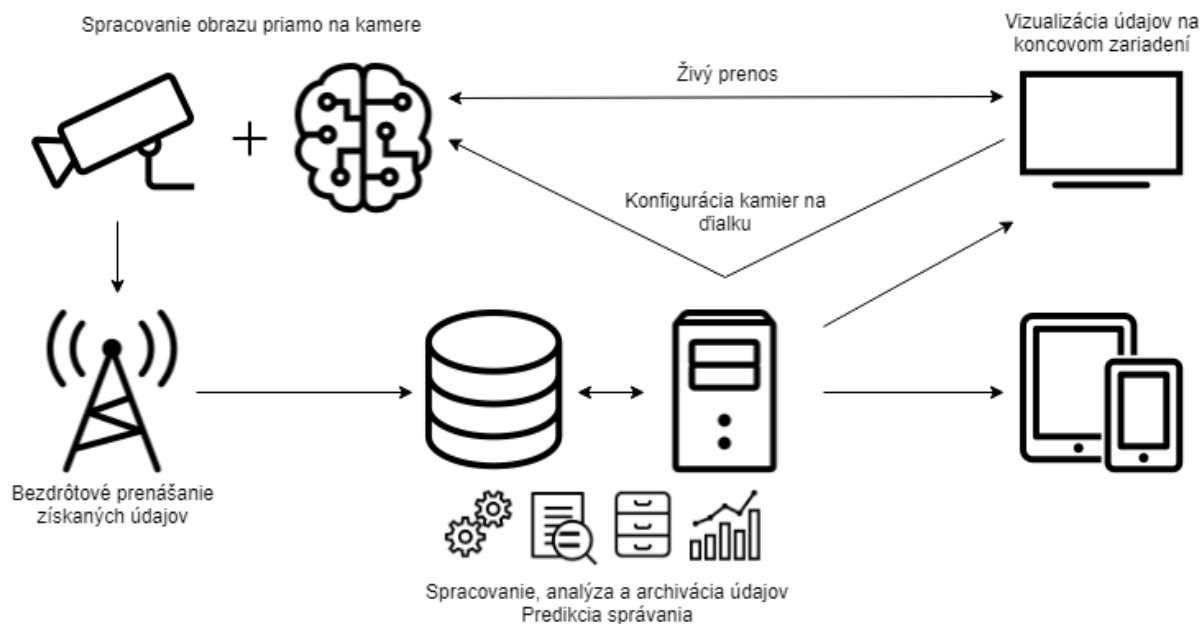
Monitorovanie oblasti zabezpečuje kamera spojená s výpočtovým modulom, ktorý zaznamenaný obraz spracúva a použitím knižnice OpenCI v ňom rozpoznáva objekty záujmu, napr. vozidlá, cyklistov či chodcov. Po rozpoznaní objektu ho kamera sleduje a zaznamenáva o ňom informácie. Medzi základné informácie, ktoré určujeme, patrí pohyb objektu medzi stanovenými zónami, jeho rýchlosť, doba státia a podobne. Jednotlivé objekty kamera taktiež vo vhodnom momente klasifikuje pomocou knižnice TensorFlow. Cieľom je spracovať video záznam priamo na mieste a neprenášať ho zbytočne cez sieť. Zistené informácie o pohybe a správaní sledovaných objektov sa odošlú na server za použitia protokolu MQTT.

Serverová časť zbiera informácie z jednotlivých kamier a priebežne aktualizuje štatistiky. Taktiež sa zaoberá dátovou analýzou, pričom využíva kombinované dáta získané z viacerých kamier. Takýmto spôsobom dokáže zisťovať informácie o toku dopravy či sledovať počet áut v danej oblasti. Server zároveň poskytuje služby vo forme REST API pre webovú aplikáciu. Server je postavený na frameworku Spring a na ukladanie dát je použitý databázový server PostgreSQL s rozšírením TimescaleDB pre efektívnejšiu prácu s časovými údajmi.

Webová aplikácia vytvorená za pomoci frameworku React sa stará o vhodné zobrazenie získaných informácií koncovým používateľom. Informácie o doprave možno zobrazovať ako text, graf alebo priamo na mape. Výzvou je nájsť spôsob, ktorý umožní zobraziť komplexné informácie o správaní sa dopravy jednoduchým a zrozumiteľným spôsobom.

3.3 Model údajov

V tabuľke 1 sú stručne opísané základné entity, ktoré náš systém používa.



Obr. 1: Schéma riešenia

Device	Fyzické zariadenie, kamera
MonitoredArea	Oblasť sledovaná kamerou
Zone	Mnohouholník definovaný pomocou súradníc vrcholov/pixelov/bodov (x,y), používaný pri sledovaní prechodu
Pass	Vstup sledovaného objektu do zóny a výstup zo zóny
DetectionParameters	Všetky konfigurovateľné hodnoty použité pri detekcii objektov kamerou (zóny, veľkosť sledovaného objektu)
Transit	Predstavuje vektor prechodov sledovaného objektu cez zóny
ZoneSequence	Predstavuje možnú kombináciu prechodov cez zóny, kvôli rýchlejšej práci s databázou
AreaStatistics	Agreguje informácie o počte áut, ktoré prešli cez prejazd za 1 hodinu
Screenshots	Ukladá posledný snímok sledovanej oblasti, po vyžiadaní nového sa starý prepíše, teda neukladáme históriu
AggregatedTrajectories	Agreguje informácie o trajektóriách sledovaných objektov, slúži na optimalizovanie vykresľovania
AggregatedMovements	Agreguje informácie o polohe sledovaných objektov, slúži na optimalizovanie vykresľovania

Tabuľka 1: Opis dátových entít

Detailnejšia dokumentácia databázy je automaticky generovaná a obsahuje detaily o jednotlivých tabuľkách a ich stĺpcoch. Tiež popisuje vzťahy medzi tabuľkami vo forme diagramu. Rovnako je dostupná aj technická dokumentácia k serverovej a frontendovej časti. Všetky sa nachádzajú v samostatne priložených dokumentoch a ich zoznam je uvedený v zozname príloh na konci dokumentu.

3.4 Repräsentácia pomocou WBS diagramu

V tejto sekcii sú znázornené WBS diagramy jednotlivých častí projektu. Pri tvorbe WBS sme použili techniku rolling wave planning, teda časti, ktorým sa zatiaľ nevenujeme, ostávajú bližšie nerozvinuté.

Obrázok 2 predstavuje legendu komponentov, ktoré sú použité pri tvorbe WBS diagramu.



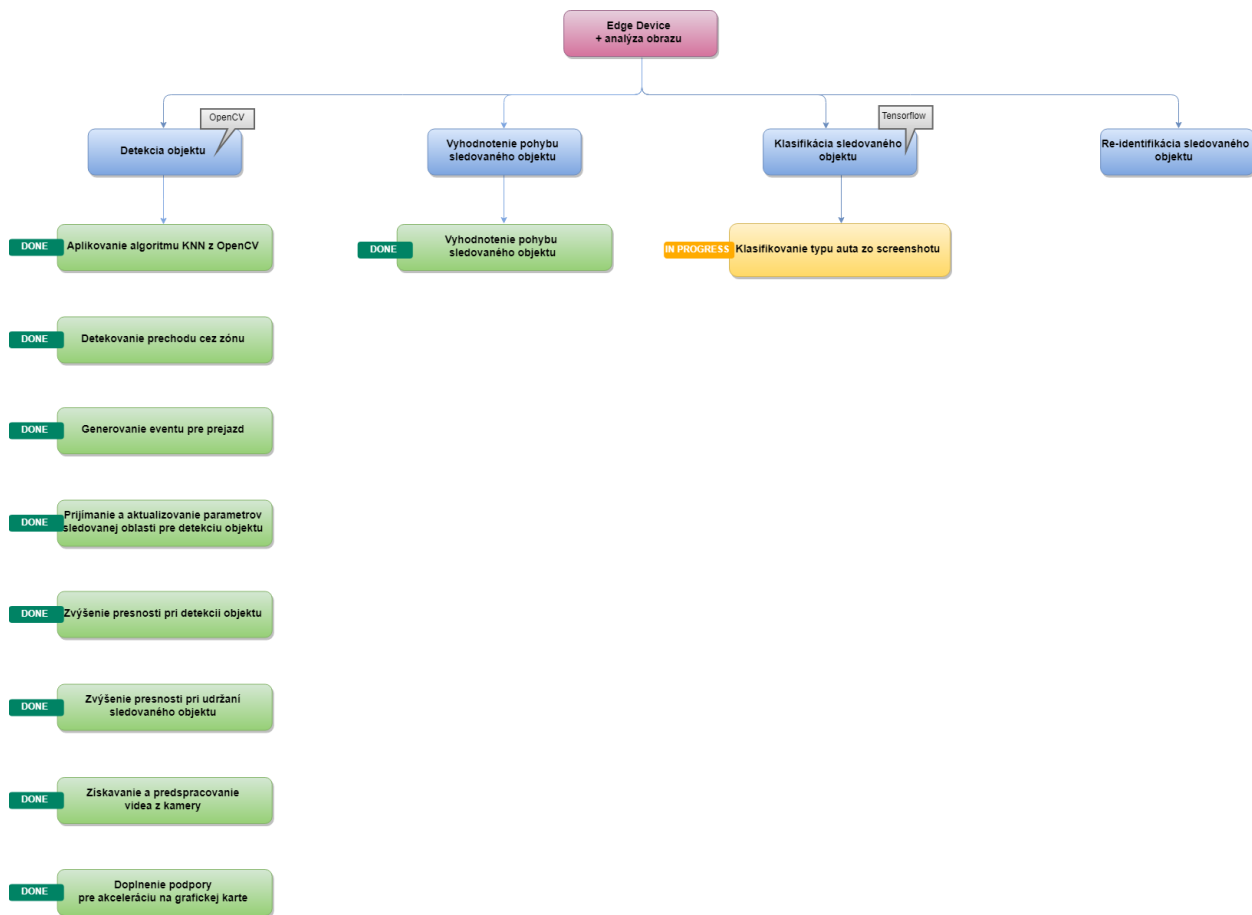
Obr. 2: Legenda komponentov použitých vo WBS

Obrázok 3 predstavuje tému *Edge Device + analýza obrazu*, ktorá sa venuje spracovaniu obrazu na kamerových zariadeniach. Táto téma obsahuje 4 epicy:

1. **Detekcia objektu.** Obsahuje predovšetkým úlohy spojené so spracovaním obrazu ako predspracovanie obrazu, detekciu objektov a prejazdov, zvyšovanie presnosti detekcie a akceleráciu na grafickej karte. Používa sa tu knižnica na spracovanie obrazu OpenCV.
2. **Vyhodnotenie pohybu sledovaného objektu.** V rámci tohto epicu sa riešilo pravidelné ukladanie charakteristík pohybu sledovaných objektov ako rýchlosť alebo smer pohybu.
3. **Klasifikácia sledovaného objektu.** Klasifikácia sledovaných objektov sa vykonáva pomocou knižnice Tensorflow a je vo vývoji. Pracuje sa na prototypu, ktorý bude integrovaný do kamerového zariadenia, schopného klasifikovať sledované objekty do kategórií auto, autobus či motorka.
4. **Re-identifikácia sledovaného objektu.** Otázkou re-identifikácie sledovaného objektu medzi viacerými kamerami na základe výrazných črt objektu či jeho pohybu sme nezačali riešiť.

Obrázok 4 predstavuje tému *Manažment*, ktorá sa venuje správe systému a konfigurácii kamier. Táto téma obsahuje nasledovné epicy:

1. **Nastavovanie parametrov pre kameru.** Konfigurovanie parametrov detekčných algoritmov na kamere na diaľku je dôležitou časťou nášho riešenia. Vyriešili sme automatické posielanie parametrov po ich zmene a pridali možnosť sledovať živý prenos z kamery. Živý prenos je oanotovaný a umožňuje sledovať, ako sa zmeny parametrov prejavili na detekcii objektov.
2. **Anotovanie sledovanej oblasti.** Aby mohla kamera správne zaznamenávať prejazdy, musí byť sledovaná oblasť anotovaná. Používatelia majú možnosť na mape alebo obrázku z pohľadu kamery definovať zóny, medzi ktorými sa budú zaznamenávať prejazdy. Patria sem všetky úlohy spojené so získaním obrázku z pohľadu kamery, ukladaním a prácou s definovanými zónami.



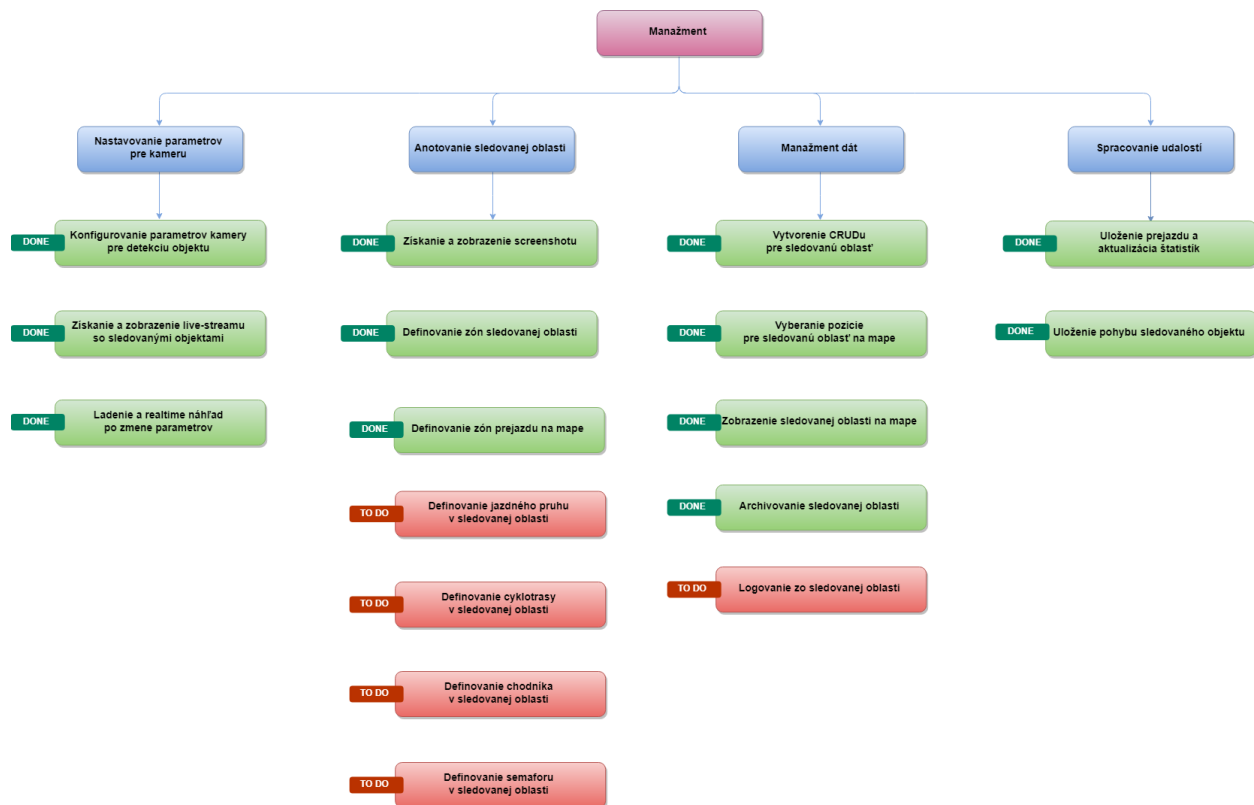
Obr. 3: Časť WBS týkajúca sa spracovanie obrazu (kamera)

3. **Manažment dát.** Patria sem úlohy spojené s CRUD operáciami nad všetkými entitami systému.
4. **Spracovanie udalostí.** V rámci spracovanie udalostí sa riešilo prijímanie a spracovanie udalostí o sledovaných objektoch z kamery. Udalosti o prejazdoch sú ukladané jednotlivo, a tiež automaticky agregované do štatistík.

Obrázok 5 predstavuje tému *Presentation*, zaoberajúcu sa prezentáciou získaných dát. Táto téma obsahuje nasledovné epicy:

1. **Štatistiky, vizualizácia z jednej sledovanej oblasti.** Obsahuje všetky úlohy spojené so zobrazením údajov z jednej sledovanej oblasti. Používateľ má k dispozícii tabuľky a grafy zobrazujúce jednotlivé prejazdy. Prejazdy taktiež zobrazujeme na mape alebo v obrázku z pohľadu kamery.
2. **Štatistiky a vizualizácie z viacerých sledovaných oblastí.** Použitie získaných údajov na vizualizáciu prepojenia viacerých sledovaných oblastí sme neriešili.
3. **Vizualizácia v obraze.** Informácie o pohybe sledovaných objektov sme využili na zobrazenie heatmapy rýchlosti, či ich presných trajektórií. Zobrazenia poskytujú možnosť filtrácie, takže používateľ má možnosť sledovať miesta, kadiaľ sa objekty pohybujú najpomalšie či najrýchlejšie.

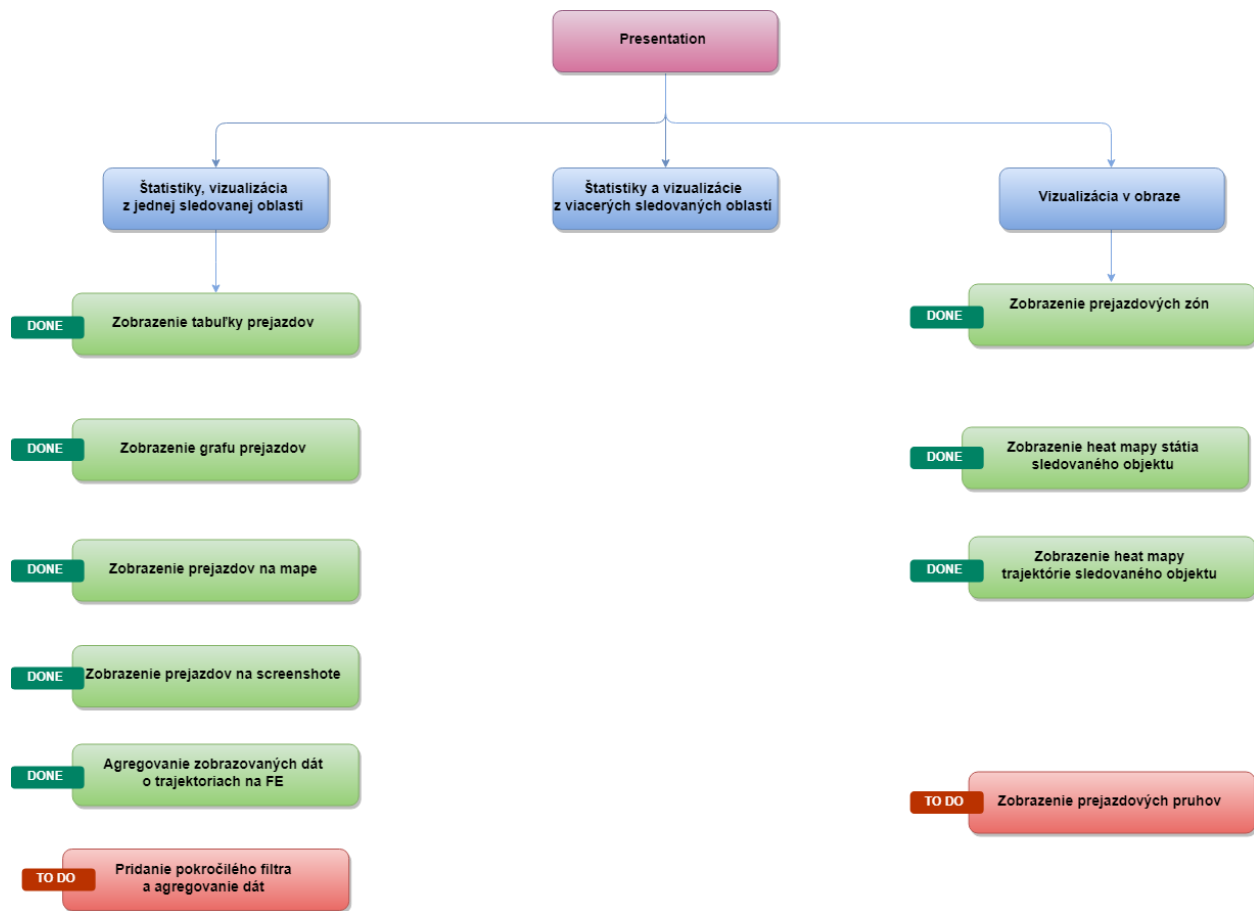
Obrázok 6 znázorňuje základné činnosti spojené so štartom projektu.



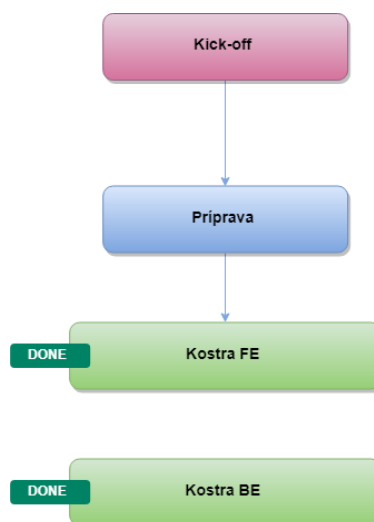
Obr. 4: Časť WBS týkajúca sa manažmentu dát (server)

4 Architektúra nasadenej serverovej časti

Nasadený backend sa líši od vývojového prostredia prítomnosťou serverov, ktoré beh aplikácie podporujú. Pre backend je nainštalovaná databáza postgresql s rozšírením timescaledb. Pre backend aj frontend je nainštalovaný a nakonfigurovaný webový server nginx. Pre backend slúži ako reverzný proxy na zabezpečenie pomocou TLS. Pre frontend hostuje samotné súbory zbuildovanej react aplikácie. Ďalej sa na serveri nachádza MQTT server mosquitto, na ktorý sa napájajú kamery a backend. Nakoniec nasadenie zahŕňa inštaláciu TURN servera coturn, ktorý sprostredkuje komunikáciu medzi kamerou a prehliadačom pri livestreame v prípade, ak by inak nevedeli nadviazať spojenie.



Obr. 5: Časť WBS týkajúca sa prezentácie dát (frontend)



Obr. 6: Časť WBS týkajúca sa štartu projektu

A Zoznam príloh

1. Dokumentácia Javadoc (backend)
2. Dokumentácia Swagger (backend)
3. Dokumentácia SchemaSpy (databáza)
4. Dokumentácia JSDoc (frontend)