



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**VIZUALIZACE ANALÝZY CHOVÁNÍ DAVU NA VIDEO-
DATECH Z DRONU**

VISUALIZATION OF CROWD BEHAVIOR ANALYSIS USING DRONE VIDEODATA

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ZDENĚK DOBEŠ

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. DAVID BAŽOUT

BRNO 2022

Zadání bakalářské práce



Student: **Dobeš Zdeněk**
Program: Informační technologie
Název: **Vizualizace analýzy chování davu na videodatech z dronu**
Visualization of Crowd Behavior Analysis Using Drone Videodata
Kategorie: Uživatelská rozhraní

Zadání:

1. Seznamte se s úlohou analýzy chování davu a specifiky videodat z dronu.
2. Na základě uživatelských požadavků připravte návrh UI vizualizující výstupy analýzy chování osob do mapového podkladu.
3. Prozkoumejte existující front-end frameworky a vyberte vhodnou technologii.
4. Proveďte návrh řešení s přihlédnutím ke zvolené technologii.
5. Implementujte prototyp aplikace.
6. Otestujte UI na reálných úlohách, proveďte optimalizace a vytvořte demonstrační video.

Literatura:

- Steve Krug: Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability, ISBN: 978-0321965516
- Tidwell et al.: Designing Interfaces: Patterns for Effective Interaction Design, O'Reilly, 2020
- William S. Vincent: Django for Beginners: Build Websites with Python and Django, ISBN: 978-1735467207
- Bradski, G. & Kaehler, A., 2008. Learning OpenCV: Computer vision with the OpenCV library, Reilly Media, Inc.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1, 2, 3, 4 a částečně bod 5.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Bažout David, Ing.**

Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.

Datum zadání: 1. listopadu 2021

Datum odevzdání: 11. května 2022

Datum schválení: 1. listopadu 2021

Abstrakt

Ústřední myšlenkou této práce je navrhnout a implementovat uživatelské rozhraní pro vizualizaci videodat z dronu. Dokument se skládá z procesu vyhotovení návrhu až po jeho implementaci do podoby webové aplikace. Využívá aplikačního frameworku Django, prezentačního frameworku React a řady aplikačních rozhraní s dominantním zastoupením Google Map API. Klade si za cíl vytvořit intuitivní prostředí pro monitorování sektorů drony a poskytnout prostředky k jeho usnadnění.

Abstract

The purpose of this work is to design and implement an user interface for visualizing videodata from a drone. The document consists of a description of the construction of a design until its implementation in the form of a web application. It uses the Django application framework, the React web framework and variety of APIs with Google Map API at the top. It aims to create an intuitive environment for monitoring sectors on a map with drones and offers useful tools to make it easier.

Klíčová slova

datové anomálie, dron, Python, Django, React, uživatelské rozhraní, HTML, CSS, Javascript, stream, video na vyžádání, vizualizace, homografie, projektivní transformace

Keywords

crowd anomaly, drone, Python, Django, React, user interface, HTML, CSS, Javascript, stream, video on demand, visualization, homography, projective transformation

Citace

DOBEŠ, Zdeněk. *Vizualizace analýzy chování davu na videodatech z dronu*. Brno, 2022. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. David Bažout

Vizualizace analýzy chování davu na videodatech z dronu

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Davida Bažouta. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....

Zdeněk Dobeš

6. května 2022

Poděkování

Velice děkuji panu Ing. Davidu Bažoutovi za poskytnutí odborných znalostí formou přátel-
sky vedených konzultací a jeho časové investice. Dále chci poděkovat všem zúčastněným na
testech uživatelské přívětivosti za ochotu a přínosnou zpětnou vazbu.

Obsah

1	Úvod	2
2	Návrh řešení	3
2.1	Specifikace požadavků	3
2.2	Analýza existujících řešení	5
2.3	Návrh uživatelského rozhraní	8
2.4	Testování uživatelského rozhraní	11
3	Analýza dostupných technologií	15
3.1	Back-end frameworky	16
3.2	Front-end frameworky	17
3.3	Shrnutí	19
4	Implementace	20
4.1	Serverová část	21
4.2	Klientská část	25
4.3	Testování aplikace	33
5	Závěr	34
	Literatura	35

Kapitola 1

Úvod

Psychologie davu je v dnešní době již velice podrobně zmapovaný fenomén, ke kterému dochází při uskupení velkého počtu lidí se společným přesvědčením či cílem. Síla anonymity a sounáležitosti pak dokáže vyvolat i v normálně pasivnějších jednotlivcích agresivnější až výtržnické chování. Přestože dokážou početnější skupiny lidí představovat jistá rizika, zaujímá právo shromažďovat se důležitou roli v Listině základních práv a svobod, mimo jiné za účelem poskytnutí možnosti protestovat, jakožto svobodného projevu nesouhlasu prostřednictvím demokratického procesu [43].

Zajištění bezpečnosti okolního majetku a přihlížejících, ale i samotného davu mají často na starosti bezpečnostní složky, které musí na celou situaci dohlížet a v případě nepokojů zasáhnout. Nebezpečí nepředstavují pouze již zformované shluky lidí, ale také místa s velkou pravděpodobností jejich utvoření jako např.: sportovní utkání. Pro zefektivnění monitorování těchto událostí se začali s rapidním technologickým posunem využívat moderní zařízení s vysokou pohyblivostí a poskytujícím přenosem video dat, drony. Tyto malé ovladatelné stroje nám umožňují sledovat větší sektory z ptáčích perspektivy, čímž redukují vynaložený čas i náklady bezpečnostních složek.

Správa video dat dronů nemusí být vždy pro dohlížejícího jednoduchá a intuitivní, což si klade za cíl výsledek této práce. Řešení umožňuje kamery těchto bezpilotních letadel centralizovat, lokalizovat a za pomoci analýzy chování davů částečně v kontextu celého pozorovacího procesu automatizovat s následným upozorněním uživatele. Monitorované území zobrazuje webová aplikace na mapovém podkladu, která jednotlivé sledované úseky dává do kontextu k zachyceným video datům, a tak poskytuje možnost trasování nepokojů způsobené davem přímo na mapě. Pro jejich autentické vyobrazení se využívá principů projekční transformace za pomoci relativních souřadnicových systémů, která mapuje kamerou zachycené zeměpisné objekty do mapy, jež aplikace využívá.

Kapitola 2

Návrh řešení

Před vyhledáním potřebných technologií a samotnou implementací představuje důležitý mezník dekompozice problému, seznámení se s existujícími řešeními a vyhotovení návrhu. Ten je vypracován na základě vstupních požadavků uživatele, zavedených konvencích a dostupných prostředcích. Po ustanovení základních vlastností designu a vnitřní logiky aplikace, se vyhotoví prototyp, který odehraje důležitou roli při testování struktury a intuitivnosti výsledného produktu. Na základě dat z výsledků testů se zredukují nedostatky návrhu, jehož modifikací se následně ocitne ve své finální podobě.

2.1 Specifikace požadavků

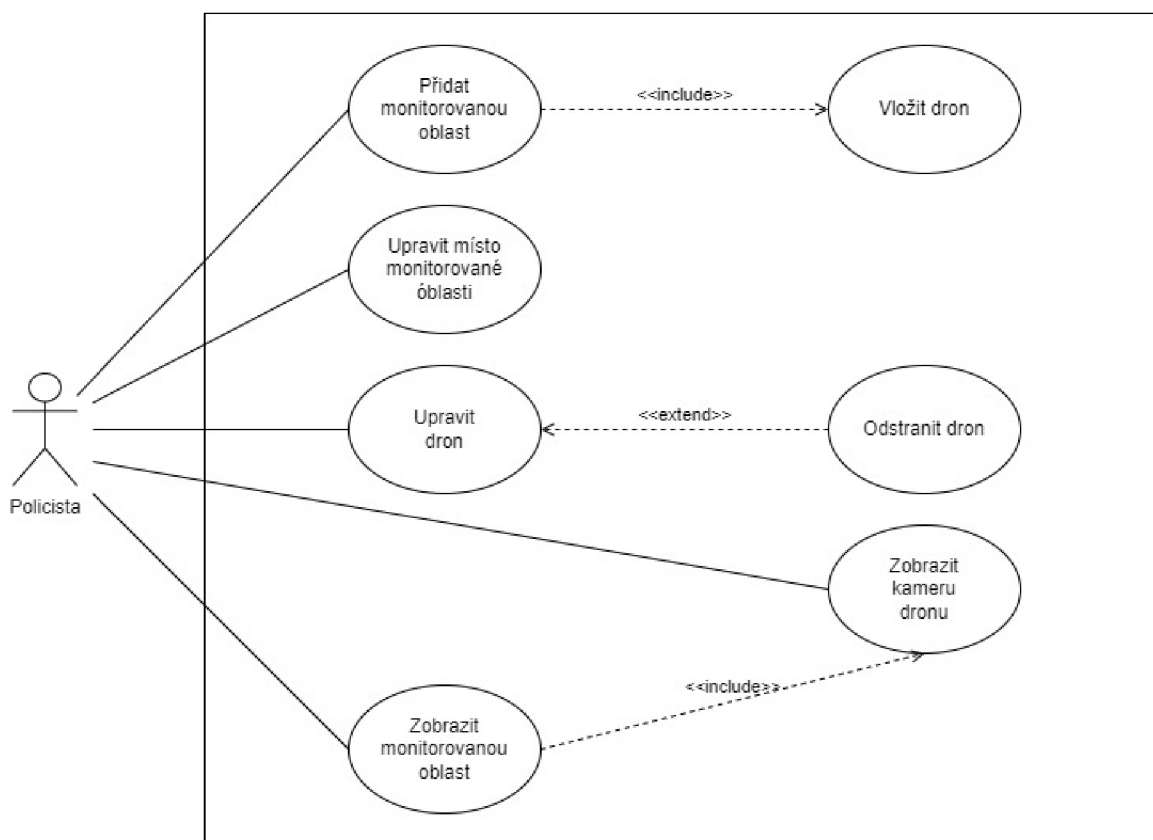
Jeden z hlavních zprostředkovatelů informací představoval zadávající práce Ing. David Bažout, na jehož programu na detekci anomálií v chování davu ve video datech z dronu[3] staví detekční logika budoucí aplikace. Výsledný produkt jeho práce výskyt případných davových anomálií zintenzivní červenou barvou na heatmapě¹, což reprezentuje neočekávanou aktivitu, a tedy i šanci na ohrožení veřejného pořádku a bezpečnosti. Program byl vytvořen ve skriptovacím jazyce Python a podporuje vlastnosti jako stabilizace obrazu a zpětné dohledání ztracené scény[3]. Na vstupu přijímá video data z dronu, které na výstupu přetváří do podoby heatmapy.

Mezi hlavní požadavky této práce patří zasazení grafické podoby heatmapy do mapového podkladu pro reprezentaci anomálie[23] na konkrétní geografické lokaci. To by v případě skutečných nepokojů mělo uživatele upozornit a umožnit mu na celou situaci pohotově reagovat. Aplikace by měla umožnit zadat vstupní informace o dronu, jako streamovací odkaz na přenášená video data, v rámci jeho inicializace. Očekává se podpora více než jednoho zařízení a zachování informací i o zařízeních neaktivních či dříve používaných. Drony se nad hlídanou oblastí aktivně nepohybují a od začátku analýzy tvoří statický obraz. Nelze od nich očekávat sledování oblasti ve vertikálně kolmém směru ani moderní technologie typu GPS. Dále se v případě poškození, nevyužití či odstavení dronu očekává možnost jeho odebrání z vnitřní paměti aplikace.

Předpokládá se různé možnosti volby reprezentace dat monitorovaných oblastí a jejich znatelné odlišení pro koncového uživatele. Žádoucí je možnost instantního přesměrování ke specifickému místu, ať už prostřednictvím blízké signifikantní socio-geografické lokality nebo již známým prostředím s nastaveným dronem a aktivním tokem vysílaných video dat.

¹heatmapa - vizuální reprezentace intenzity na základě barvy

V případě kolize monitorovaných oblastí na mapovém podkladu musejí být úseky viditelně odlišené, ale zároveň obsahově čitelné.



Obrázek 2.1: Diagram případů užití.

Nástroj by měl dokázat reprezentovaná video data na mapě zobrazit i v jejich nezkreslené podobě klasického živého přenosu. Užitečnou se jeví i ústřední vlastnost videí na vyžádání, a to funkce posunu v čase pro vyobrazení již proběhlé davové anomálie k vyhotovení analýzy a odhalení podnětu řetězce nadcházejících událostí.

Webové rozhraní by mělo být přívětivé především pro uživatele tabletových zařízení, kvůli jejich začlenění do vybavení bezpečnostních složek v terénu, ale také použitelnost na běžném širokoúhlém monitoru. Očekává se srozumitelnost bez nutnosti znalosti odborné terminologie. Nakonec se předpokládá elementární znalost anglického jazyka uživatelů, tudíž se v rámci rozšíření cílové skupiny využije právě ten.

Souhrn požadavků

Specifický požadavek	Minimální požadavky	Žádoucí vlastnosti
Cílová zařízení	Tabletové zařízení	Podpora pro stolní počítač
Pozadí	Mapový podklad	Různé typy zobrazení
Správa dronů	Inicializace dronu	Modifikace a odstranění
Monitorovaná oblast	Transformace video dat do mapového podkladu	Odlišení sektorů a přesměrování k nim
Kamera dronu	Zobrazení živého přenosu	Posun v čase
Autentizace	Přihlášení, registrace	Správa účtu

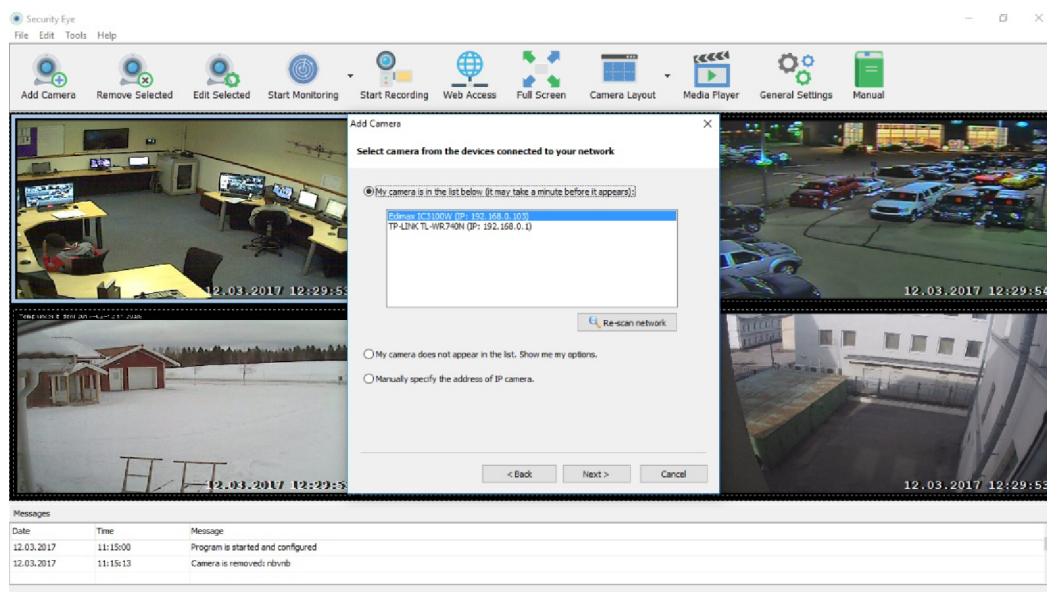
2.2 Analýza existujících řešení

V oblasti monitorování lokací existuje spousta volně dostupných programů, které často sdílí podobnou vnitřní strukturu a design. Většina však zakládá na znalosti konkrétních sledovaných sektorů, neboť neposkytují žádný kontext k umístění kamerového zařízení a nabízejí pouze základní funkcionalitu. Aplikace zaměřené specificky na správu dronů, mnohdy s komerčním charakterem, skýtají naopak implementaci složitější, avšak obvykle hardwarově omezenou nebo až příliš provázanou s ovládáním dronu.

Výběr analyzovaných produktů proběhl na základě designu, popularity a komerčního využití. Apelováno bylo také na selekci takových řešení, jež dle personálních kritérií zpracovávají specifické problematiky nejelegantněji, či inovativně a intuitivně.

Security Eye

Každý člověk někdy pocítil strach z poškození či odcizení svého majetku. Právě tito lidé jsou jedni z cílových uživatelů aplikace Security Eye[37]. Tento neplacený program poskytuje kromě základní správy kamer i pokročilé techniky řešení detekce pohybu, jako automatické screenshotování nebo zachování záznamů podezřelých částí přenosu. Jeho grafický design hlavního menu se skládá z nejpoužívanějších funkcí reprezentované ikonkami a textovým popisem 2.2. Cílová zařízení představují počítače s operačním systémem Windows, tudíž lze předpokládat, že uživatelé s přístupem pouze k mobilnímu telefonu se nejčastěji dozví o nekonzistencích svého majetku až doma.



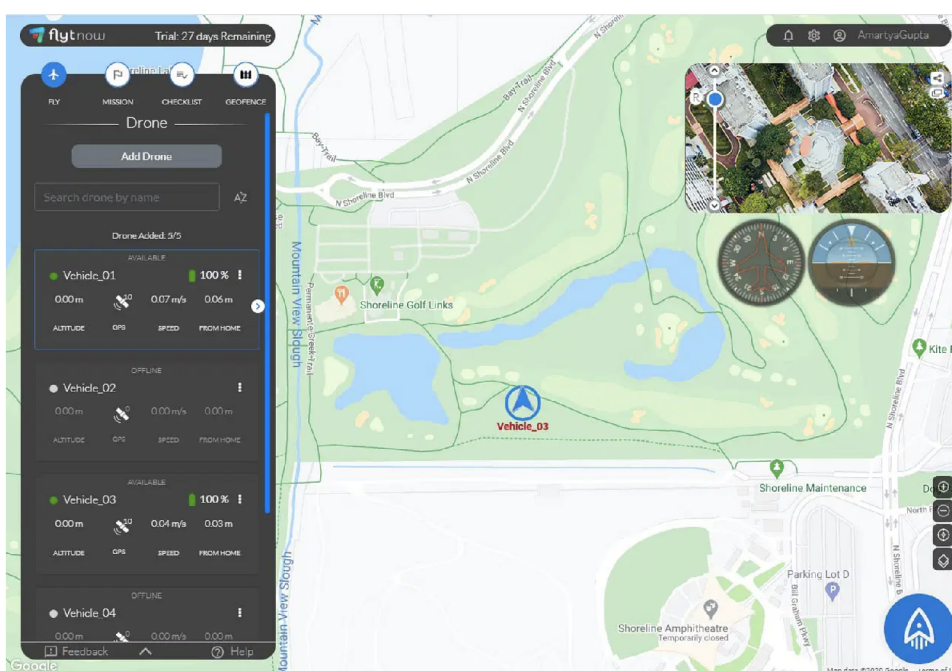
Obrázek 2.2: Uživatelské rozhraní aplikace Security Eye (převzato z [37]).

Inspirovat se lze především hlavní nabídkou, simplifikovaným přidáváním kamerových zařízení a jeho správou, která uživatele odlišuje od pokročilé technologické terminologie. Za zmínku stojí i explicitní odstartování monitorování, jehož neexistence by mohla způsobit nadbytečnou délku pořízených video záznamů. Kromě sledování objektů kamerovými systémy prostřednictvím separovaných oken v reálném

časem má aplikace k dispozici také možnost posunu zpátky v časem pro restaurování² zachycených událostí. Poslední relevantní vlastností představuje nastavení a podpora v podobě manuálu.

FlytNow

Pokud jste nadšenec do automatizace dronů či pověřen zabezpečením heterogenních oblastí, aplikace FlytNow[16] si najde vaši přízeň. Toto multiplatformní komerční dílo umožňuje vzdáleně předdefinovat trajektorie dronů a v nastavených intervalech je opakovat, čímž dokáže pokrýt i rozmanitější sektory. Osobitost si najde především v moderním designu a množstvím dostupných funkcí. Hlavní menu se upíná po celou dobu chodu aplikace na levé straně a využívá občasných pop-up vycentrovaných oken pro příležitosti operace jako přidání dronu nebo podrobnější popis zařízení.



Obrázek 2.3: Uživatelské rozhraní desktopové aplikace FlytNow (převzato z [16]).

Jelikož se požadavky na očekávanou funkcionalitu uvedené v kapitole 2.1 ztátně shodují s těmi u analyzované aplikace, jedná se o nezanedbatelný zdroj informací o nadcházejícím vyhotoveném řešení. Především tedy reprezentuje konvence, které mohou uživatelé dronů očekávat. Při prvním pohledu na uživatelské rozhraní zaujme hlavní menu, jenž může mít kvůli jeho statickému markantnějšímu charakteru negativní efekt na dostupnou velikost sledovaného prostoru. Ve výchozím nastavení se v něm nachází list dronů, který o jednotlivých zařízeních obsahuje nejdůležitější informace a jejich kontext. Při stisknutí dronu se úsek mapy přesměruje na jeho lokaci a zobrazí aktuální vysílaná video data.

Prvky práce s mapou se nacházejí v pravém dolním rohu, což se rozchází s konvencemi aplikací velkých korporací jako Google. Kromě tradičního oddálení a přiblížení

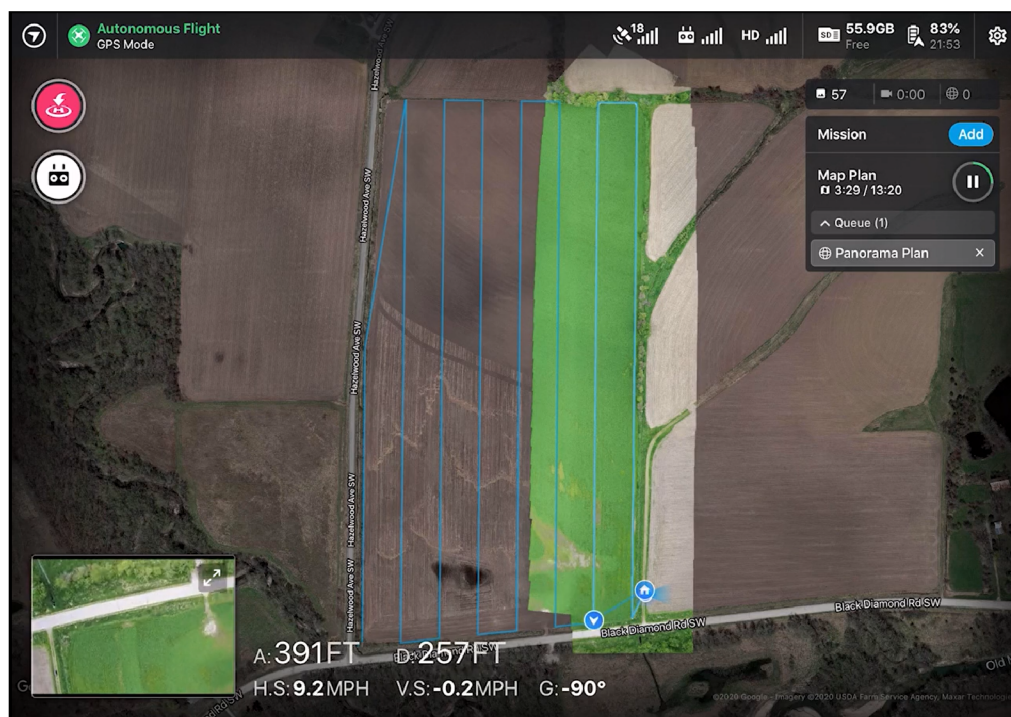
²restaurování - znovuobnovení či uvedení do původního stavu

se v jejich řadách vyskytuje tlačítko přepnutí mezi vrstvami jako turistická či satelitní. Velikost těchto i dalších ikoněk by mohla při práci na tabletu tvořit komplikace. Přesunovat se na mapě lze za použití myši a pravděpodobně kvůli hardwarové podpoře GPS není implementovaná možnost přesunu na lokaci prostřednictvím textového pole.

V neposlední řadě aplikace nabízí možnost manuálního nastavení ohraničeného sektoru určeného k monitorování. Tvar lze definovat za pomoci manipulace s výchozími hraničními body polygonu. Oblast, nacházející se v ně takových objektů, se na mapě zbarví do průhledné zelené. Kraje naopak červeně kvůli kontrastu s nejčastěji používanými barvami pozadí, a to modré, zelené a šedé. Tímto se lze inspirovat v rámci požadavku zobrazování kamery na mapovém podkladu.

DroneDeploy

Společnost stejnojmenné aplikace DroneDeploy[14] představuje jednu z předních špiček ve zpracování dat prostřednictvím dronů. Jejich produkt dokáže za pomoci fotek pořízených ze vzdušného prostoru vypracovat 3D modely lokací, zjistit kvalitu úrody nebo třeba vypracovat stavební výkres již postavené budovy. Pro získání dat stačí nastavit pár inicializačních informací jako okraje zkoumané oblasti a výška letu pro bezpečí dronu i okolí, a aplikace se za autonomní práce bezpilotního letadla postará o zbytek. Určena je především pro komerční účely.



Obrázek 2.4: Pořízení fotek oblasti ke zpracování ve streamovacím režimu aplikace DroneDeploy (převzato z [14]).

Jakožto druhý reprezentant úspěšné aplikace zpracovávající data z kamer dronů, slouží tento produkt jako důležitý zdroj informací. Před vytvoření projektu se uprostřed nahoře ukáže uživateli vyhledávací pole pro přesměrování k dané destinaci.

Následně uživatel pohybem po mapě upřesní lokalitu a začne pracovat s hlavní nabídkou, jež je rozklikávací z důvodu maximalizace viditelné plochy a nachází se v levém horním rohu.

Pozastavit se lze především nad vykreslením zpracovaných oblastí ve formě 3D modelu na mapovém podkladu 2.4. Tyto záznamy pak tvoří dokonalý kontrast s dosud nezachyceným prostředím a vytvoří uživateli obrázek o výsledném díle ještě před vypracováním. Model se vyhotovuje za pomoci systému VPS, který dokáže lokalizovat jemu nejbližší objekt na povrchu díky kombinaci ultrazvukových senzorů a technologie optického toku [36]. Při užití funkcí jako zobrazení nadmořské výšky se model zbarví ve formě heatmapy. Ostatní prvky aplikace jsou buď nepodstatné nebo podobné u řešení z kapitoly 2.2.

2.3 Návrh uživatelského rozhraní

Ve veřejném povědomí všech designerů je zahrnuto pořekadlo, že kvalitní návrh šetří peníze i čas, neboť jeho řádné vypracování minimalizuje počet nejasností a vad ve finálním produktu. V rámci celého návrhářského procesu se nejprve iterativním postupem [42] vypracovala počáteční verze uživatelského rozhraní a při každé další iteraci byl její aktuální stav konzultován a diskutován o jeho omezeních s jednotlivými alternativními úpravami. Následovalo vyhotovení testovacího protokolu, jenž měl návrh podrobit testem použitelnosti [25]. Specifikace protokolu a detaily testu jsou popsány v kapitole 2.4. Testovaná verze se na základě výsledků modifikovala do finální podoby, která slouží jako podklad pro implementaci koncového produktu.

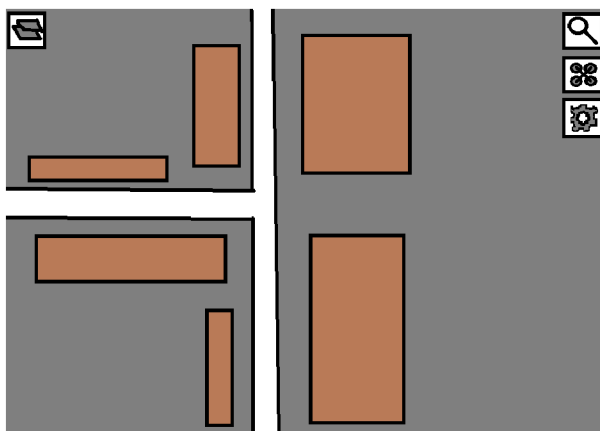
Návrh si klade za cíl vytvořit známé, ale částečně unikátní prostředí, aby uživatelé příliš nezatěžovali učící křivkou a zároveň se dostatečně dokázalo odlišit od ostatních aplikací. Slouží především jako kostra, jež není ještě kompletně barevně a stylisticky unifikována. V návrhu není zakomponovaná podpora pro tělesně postižené, kvůli vstupnímu testu zdravotní způsobilosti do řad policie [28]. Ze stejného důvodu se netolerují výrazné poškození či nemoci oka a nedostatečné barevné vidění základních barevných signálů červené, žluté, modré a zelené. V potaz se neberou ani další majoritní duševní poruchy nebo poruchy chování.

Logická struktura

Stejně jako již zmíněná existující řešení popisována v kapitole 2.2, tak i v této aplikaci představuje hlavní složku a prostředek manipulace mapový podklad. Ten se rozprostírá po celé ploše okna a zastřešuje konvenční operace jako pohyb tažením nebo standartní metody přiblížení a oddálení. Vyjímaje mapy, oplývá domovská stránka také tlačítky s vrstvami mapy a pomyslným postranním panelem reprezentující hlavní nabídku 2.5.

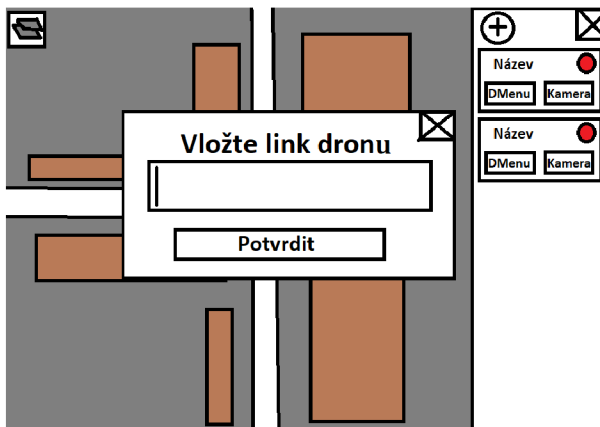
Mapové vrstvy, umístěné v konvenční lokalitě levého horního rohu, umožňují přepínat mezi režimem zobrazení čisté rekonstrukce video dat z dronů v mapovém podkladu a zobrazením jejich heatmap s výchozím thresholdem. První režim slouží pro aktivní monitorování sledované situace a druhý je zaměřený spíše na inaktivní pozorování často kombinované s další odlišnou činností pro zvýšení efektivity policisty. Postranní panel, který se rozprostírá v pravém horním rohu z důvodu dominantní

pravé ruky u většinové populace, obsahuje základní orientační prvky aplikace. Panel je minimalistický z důvodu maximalizace dostupného místa pro monitorování, jež se odehrává primárně v oblasti mapovém podkladu.



Obrázek 2.5: Návrh hlavního okna aplikace.

V domovské nabídce máme k dispozici sadu funkcí reprezentovány symboly, jež jsou popsány v kapitole 2.3. Pro selekci funkcí bylo dbáno na pravidlo, kdy by se uživatel měl dostat do kterékoliv sekce maximálně v rámci 3-5 kliknutí [25]. První tlačítko ve tvaru lupy lze využít pro vyhledání specifické lokace na mapě. Po jeho stisknutí se otevře textové pole, kam se zadá hledaná destinace a potvrzením proběhne přesměrování. Následuje ikona dronu a ozubeného kola. Ozubené kolo otevře okno nastavení, kde si lze uzpůsobit chování aplikace a spravovat účet. Většina možností úprav v tomto okně budou navrženy v průběhu testování výsledného produktu na základě zpětné vazby uživatelů. Ikona dronu zobrazí správu dronů.



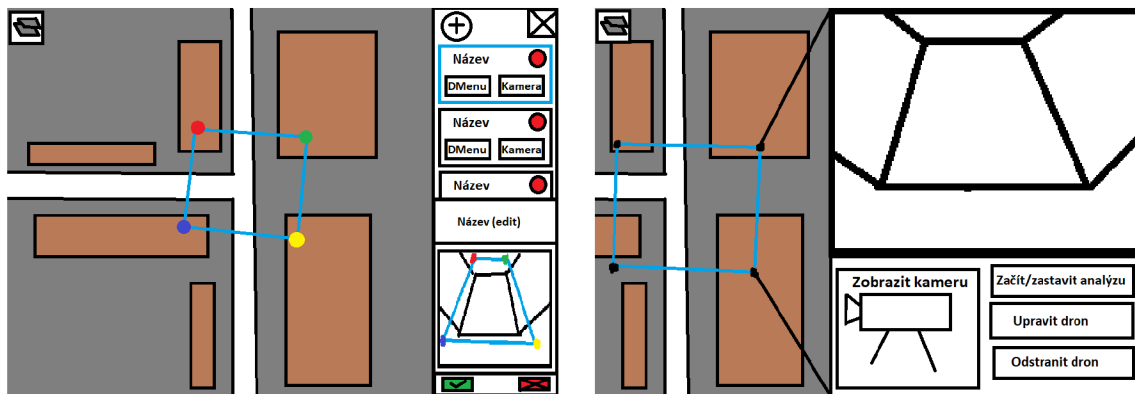
Obrázek 2.6: Přidání dronu ve správě dronů.

Správa dronů obsahuje list veškerých aktivních i neaktivních dronů, jejich názvy, odkazy na kameru a individuální menu. Jeho aktivitu lze ověřit pomocí kruhového symbolu vedle názvu, jež je při aktivní analýze video dat rozsvícen zeleně, jinak červeně. Pro vložení nového zařízení slouží tlačítko přidání v levém horním rohu listu,

po jehož stisknutí se otevře okno, které vložení streamovacího odkazu a zmáčknutím tlačítka potvrdit, vytvoří reprezentační jednotku dronu 2.6 v listu.

Před kompletní inicializací je však nutné nastavit dvě sady 4 bodů, které reprezentují skutečné objekty reálného světa nacházející se na mapě a zároveň na záběrech dronu 2.7a. Díky tomu lze aproximačně mapovat video data do mapového podkladu. Umístění bodů kamery, které se ve výchozím stavu nachází přímo v rozích, se v novém okně nastaví tažením na žádoucí pozice. Odpovídající body na mapě se ve výchozím nastavení nachází v krátké vzdálenosti a je možné je modifikovat stejným principem. Vztah jednotlivých bodů znázorňují totožné barvy. Před potvrzením se také může nastavit volitelný název dronu. Modifikovat tyto informace lze kdykoliv kliknutím na obdélníkovou strukturu dronu v listu či tlačítkem **Upravit dron** v nabídce dronu.

Pokud aplikace obsahuje alespoň jeden dron, můžeme vstoupit do jeho menu 2.7b. To je přístupné pomocí tlačítka **DMenu** ve správě dronů nebo prostřednictvím kliknutí na plochu uvnitř jeho bodové reprezentace na mapě. Po otevření dojde na mapě k přeměrování na monitorovanou oblast a otevření okna obsahující aktuální streamovaná video data v horní části nabídky. Lze tak rychle nahlédnout na živý přenos bez ohledu na mapovou vrstvu či zkreslení vyvolané transformací záznamu do mapy. Okno obsahuje tlačítko k odstartování nebo ukončení analýzy, přeměrování na nastavení bodů, odstranění dronu a možnost zobrazit kameru na celé obrazovce. Režim kamery na celé obrazovce, kterého lze dosáhnout i dvojklikem na bodový objekt na mapě, obsahuje možnost pohybu zpátky po časové ose pro rekonstrukci událostí. Daná funkce může být žádoucí v případě, že byl uživatel na anomálii upozorněn teprve prostřednictvím heatmapy nebo chce zpětně dohledat konkrétního agresora či spouštěč kaskády událostí.



(a) Nastavení polohy dronu.

(b) Menu dronu.

Estetika a symbolika

Předpokládá se, že vizuální stránka bude na základě zpětné vazby uživatelů dynamicky měněna v průběhu implementace i dodatečných úprav po testech uživatelského rozhraní, proto tato sekce slouží pouze jako inicializační hodnota, jež bude s postupem času modifikovaná do cílové podoby.

Mapový podklad oplývá standardní sadou barev typickou pro aplikace podobného typu. Jiné sady, v rámci výchozích vrstev map jako turistická či satelitní budou

implementovány na základě knihovny podpory. Nová vrstva, jež zobrazuje pouze reprezentace heatmap se bude nacházet mezi nimi. Vyskytnuté anomálie se vykreslují červeně, kvůli její unikátnosti v mapách a evoluční reprezentaci nebezpečí. Hranice bodových polygonů jsou spojovány modrou z důvodu její neutrality [10] a mizivé pravděpodobnosti užití aplikace za účelem monitorování vodních ploch.

Postranní panel hlavní nabídky obsahuje trojici symbolů počínaje lupou, sloužící k vyhledání zeměpisné lokality 2.6. Ta bude, v případě nejednoznačného výkladu její funkcionality, ozvláštněna obrázkem mapy na kraji či v centru. Dalším v pořadí je symbol dronu, představující zobrazení dronů. Pokud dojde k přidání nebo přesunutí dalších funkcí dronů do panelu, budou ikony obohaceny o specifikující symboly na okrajích. Nakonec se zde nachází tradiční ozubené kolo, vyskytující se dnes takřka ve všech softwarových produktech, jež obsahuje sekci nastavení. Barva okrajů oken i ikon bude z důvodu vybudování kontrastu vůči mapě implementována v černé.

V listu dronů se používá kombinace zelené a červené barvy pro standartní zobrazení aktivního či neaktivního zařízení, ale také u potvrzování nastavení dronů. Zařízení, se kterým se aktivně pracuje se vyobrazuje s modrým okrajem. Název jednotlivých zařízení se stylisticky liší, aby neevokoval v uživateli pocit, že ho lze stisknout [25]. Jelikož list nezabírá v kontextu celého rozhraní příliš prostoru, řeší symbol operace sčítání zasazeném v kružnici, jež je na mobilních zařízeních typický, reprezentaci ikony přidávání nového zařízení.

V nabídce dronu reprezentuje vztah okna a polygonu čtyřúhelník, který oba objekty spojuje 2.7b.

2.4 Testování uživatelského rozhraní

V rámci odchytení signifikantních nedostatků se provedly testy použitelnosti [25], ve kterých byl uživatel konfrontován s chodem aplikace prostřednictvím prototypu. Testování se zaměřovalo na kvalitativní způsob získávání informací, kvůli menšímu rozsahu aplikace a nedostatku finančních i lidských zdrojů. Kladlo si za cíl empiricky změřit efektivitu, přívětivost a intuitivnost návrhu uživatelského rozhraní. Vyhotovení prototypu proběhlo v papírové podobě a testovacímu subjektu byl při interakci s ním simulován běh aplikace adekvátními změnami vzhledu prototypu, jež manuálně vyhotovil pozorovatel.

Testovací protokol

Testování jednotlivců bylo navrženo na průměrnou délku 15 minut, zahrnujíc všechny aspekty procesu vyjímaje vyhledání místa.

Metodika

1. Nalezení optimálního místa bez externích rušivých elementů jako např.: zapnutá elektronika, nadměrný hluk, přihlížející lidé atd.
2. Seznámení subjektu s účelem testování, průběhem, smyšlenou situací a jeho úkolem. Odpovězení na dodatečné otázky.
3. Splnění úkolu v aplikaci za pomoci pozorovatele, který bude poskytovat makety pro simulování jejího chodu bez poskytování rad a názorů.

4. Subjekt i pozorovatel vyplní přichystané dotazníky, čímž se test ukončí.

Uvedení subjektu do kontextu role

POV:³ Jste policista, který má za úkol dohlížet na poklidný chod fotbalového zápasu, a v případě diváckých nepokojů zasáhnout. Nad danou oblastí máte rozmístěné dva drony s kamerami, které celou oblast monitorují. Od Vašeho staršího kolegy Vám byla doporučena aplikace, o které slyšel, že má celý proces značně usnadnit. Aplikaci si stáhnete a spustíte.

Úkoly

1. Vložte do aplikace živé přenosy obou dronů, aby jste byli schopni celou situaci pohodlně monitorovat. K dispozici máte streamovací odkazy na živé záznamy kamer dronů.
2. Aplikace vyslala upozornění o neočekávané aktivitě na stadionu. Zobrazte si jeho aktuální dění na kameře a zjistěte, co danou situaci způsobilo.

Formulář subjektu

věk, pohlaví, nejvyšší dosažené vzdělání

1. Bylo nastavování jednotlivých dronů intuitivní?
Nebylo - Spíše nebylo - Tak napůl - Spíše bylo - Bylo
2. Bylo snadné nalézt kamery dronů?
Nebylo - Spíše nebylo - Tak napůl - Spíše bylo - Bylo
3. Bylo jednoduché se v aplikaci orientovat?
Nebylo - Spíše nebylo - Tak napůl - Spíše bylo - Bylo
4. Byla aplikace zábavná na použití?
Nebyla - Spíše nebyla - Tak napůl - Spíše byla - Byla
5. Co bych na aplikaci změnil/a
-Otevřená otázka-

Formulář pozorovatele

1. Bylo nastavování jednotlivých dronů pro subjekt intuitivní?
Nebylo - Spíše nebylo - Tak napůl - Spíše bylo - Bylo
2. Bylo nalezení kamer dronů pro subjekt snadné?
Nebylo - Spíše nebylo - Tak napůl - Spíše bylo - Bylo
3. Orientoval se subjekt v aplikaci?
Ne - Spíše ne - Tak napůl - Spíše ano - Ano
4. Vykazoval subjekt při užívání aplikace známky frustrace?
Ne - Spíše ne - Tak napůl - Spíše ano - Ano
5. Kde a jaké problémy měl subjekt?
-Otevřená otázka-

³POV - Point of view

Způsob vytvoření výsledků

Jednotlivé odpovědní varianty představují číselné hodnoty počínaje číslem 1 u odpovědi nejvíce vlevo a dále se inkrementují směrem doprava až po číslo 5. Pro redukci subjektivity ze strany testovaného se otázky 1-4 mezi formuláři subjektu a pozorovatele zprůměrovaly. Numerické hodnoty problémů byly založeny na korelaci otevřených otázek s hodnotami odpovědí 1-4, na které může být nahlíženo jako na metadata chyb. Z otázek 1-2 se započítává vždy pouze 1 dle odpovídající sekce, ve které se uživatel při narazení na daný problém nachází. Nakonec se vypočítá aritmetický průměr výsledků testů, zaznamená se četnost výskytu každého problému pro upřesnění jeho závažnosti a uvede se v metadatach. Reverzní hodnota odpovědi na otázku 4 ve formuláři subjektu částečně poskytuje zpětnou vazbu o míře frustrace uživatel nad neúspěšnou lokalizací funkcí, tudíž se v rámci statistiky pracuje s ní.

Výsledky

Testem bylo podrobena 6 účastníků ve věkové skupině 20-22, jež dosáhli středoškolského vzdělání. Vybírání byli především jednotlivci se zaměřením mimo technické obory. Výsledky jejich celkové manipulace s aplikací jsou převedeny na procentuální vyčíslení z důvodu elegantnější reprezentace.

Použitelnost aplikace

Nastavování dronů	Nalezení kamer	Intuitivnost	Míra frustrace
71.6 %	83.3 %	81.6 %	13.3 %

Dále se vyseletovaly všechny problémy s větší četností než 1 nebo znatelnou korelací při hodnocení použitelnosti a byly dle metadat seřazeny sestupně od nejzávažnějších. Spodní hranice se kvůli nízkému počtu účastníků určovala částečně subjektivně.

Nejzávažnější problémy

1. Nejasná inicializace pozice dronu
2. Očekávaný automatický začátek analýzy
3. Chybějící křížek nebo tlačítko zpět v menu dronu
4. Nejednoznačnost tlačítka pro přidání dronu

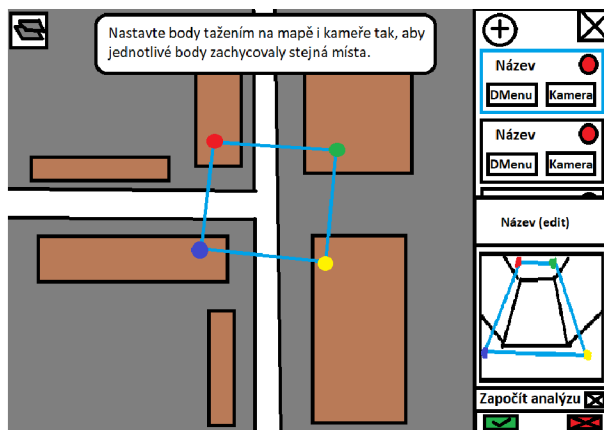
Testování splnilo očekávání a upozornilo na slabé články návrhu, především na ty spojené s nastavováním pozice dronu.

Modifikace návrhu

Na základě výsledků z testování uživatelského rozhraní dojde k modifikaci návrhu, aby zamezil nejasnostem, které starý návrh podněcoval. Úprava proběhne prostřednictvím řešení nejzávažnějších problémů z kapitoly 2.4 a bez rozsáhlejších zásahů do logické struktury aplikace.

Inicializace pozice dronu přichází s netradičním způsobem nastavování, tudíž bude neoptimálnější s ním uživatele při prvním použití seznámit formou nápovědy 2.8. Zobrazí se vždy pouze u inicializace hodnot a půjde vypnout v sekci nastavení.

Uživatelé si nejsou vědomi o všech dostupných funkcích aplikace, takže by se v rámci vytvoření dronu měla možnost spustit **automaticky analýza** sama. Ta je nyní ve výchozím stavu zaplá, aby evokovala pocit, že se jedná o žádoucí a důležitou funkci. Přednastavenou hodnotu stavu lze opět změnit v nastavení.



Obrázek 2.8: Modifikovaná verze s přednastavenou analýzou a nápovědou.

Nabídka dronu byla obohacena o **Chybějící návratová možnost** na standartním místě v pravém horním rohu a **tlačítko pro přidání dronu** bude v případě, že se v listu doposud nenachází žádné zařízení, obdařeno malou nápovědou s názvem funkce. Tím se návrh dostává do své finální podoby, jež bude sloužit jako základní podklad pro implementaci v kapitole 4.

Kapitola 3

Analýza dostupných technologií

Před samotnou implementací aplikace, představuje důležitý mezník selekce neoptimalnější technologie k jejímu vyhotovení. Standardní sadu nástrojů pro vývoj webových rozhraní tvoří jazyky `HTML`, `CSS`, `Javascript`, sloužící k vytvoření vizuální front-end části a jazyk `PHP` na sestavení logického back-endu. Ačkoliv jsou tyto základní prostředky schopné poskytnout dostatečnou podporu na vytvoření funkcionality i vzhledu webu, je z důvodu finančních a časových úspor využívána řada webových frameworků, jež si kladou za cíl celý implementační proces zjednodušit a zefektivnit.

Webové frameworky slouží jako nadstavba nad specifickým programovacím jazykem, mohou být rozčleněny na aplikační neboli back-end frameworky, zpracovávající příchozí požadavky uživatele a komunikující s databází, a na prezentační neboli front-end frameworky, jež požadavky odesílají a starají se o grafickou reprezentaci dat jejich odpovědí. Frameworky nabízejí především řadu knihoven, šablon, alternativní řešení API¹ a podporu práce s databází, čímž poskytují bezpečnost, udržovatelnost a rozšiřitelnost kódu. Často bývají zmiňovány v kontextu s dynamickými weby, které jsou specifické pro svoji modifikaci struktury `HTML` přímo za běhu [2].

Běžná kritéria výběru

- Znalost zvolené technologie programátorským týmem
- Pochopitelnost jazyka z důvodu údržby aplikace a snazšího paralelního vývoje
- Podpora nástroje ze strany jeho vývojářů
- Funkce, které daný prostředek nabízí

V rámci analýzy dojde u zkoumaných technologií k zaměření se na již zmíněné faktory a snaha o jejich objektivní ohodnocení. Vzhledem ke skutečnosti, že první dva body se pohybují v silně subjektivním spektru, bude výsledek selekce značně ovlivněn osobními preferencemi. Činitelé tohoto průzkumu implicitně zahrnují srozumitelnost a množství podpůrných zdrojů, stylistiku kódu a moji implementační historii.

¹API - Application Programming Interface

3.1 Back-end frameworky

Jelikož aplikace neklade nároky na komplexní komunikaci s databází, bude výběr aplikačního frameworku záviset hlavně na jeho API, strukturovanosti a uživatelské přívětivosti. Rešerše proběhla z řad nástrojů zakládajících se na MVC modelu, kvůli jejich aktuálnosti a urychlení procesu implementace.

Ruby on Rails

Framework Ruby on Rails (RoR), založený na skriptovacím jazyce Ruby, si našel své oblíbence již od svého založení v roce 2004 [18]. Tento open-source software se silně zaměřuje na paradigmatu CoC², jež si klade za cíl minimalizovat počet nutných konfigurací, a DRY³ neboli znovupoužívání již napsaného kódu prostřednictvím externích modulů, které nesou název gems [24]. Ty jsou tvořeny převážně vývojářskou komunitou. Mezi nejpopulárnější patří např.: Pry nebo Draper [32].

Samotný jazyk Ruby vznikl jako reakce na vlnu popularity skriptovacího jazyku Python a měl oproti němu poskytnout intuitivní, až řeči podobnou syntax. Je plně objektivně orientovaný a zaměřený primárně na práci s webem [15].

RoR zohledňuje komplexní databázové systémy, systémy pro správu obsahu a jednostránkové aplikace [24]. Další z hlavních předností představovala podpora skriptovacího jazyka SASS⁴, který dokáže zkompileovat do jazyka CSS, ta však přestala být relevantní aktualizací v roce 2019, v rámci které došlo k ukončení podpory [41].

Nevýhoda RoR spočívá v jeho nižší rychlosti a větším výkonovým nárokům na serverovou část [24]. Komplikace může také tvořit problém s nalezením kvalitnější dokumentace.

CakePHP

Webový framework CakePHP vznikl v 2005 jako reakce na úspěch RoR, jehož koncepty byl zdatelně inspirován. Drží se například již zmiňovaného paradigmatu CoC a využívá tzv. pluginů, jež lze připodobnit k RoR gemům [6]. Tento nástroj, napsaný v programovacím jazyce PHP, se pyšní jednou z nejrychlejších učících křivek, především díky své politice CRUD⁵, která je nejznámější pro své uplatnění v jazyce SQL [31], a technice objektově relačního mapování, díky níž dochází k automatické konverzi mezi relační databází a objektově orientovaným jazykem [8].

Hlavní přínos představuje zabezpečí proti standartním typům útoků jako sql injection, input validation attack, ale i CSFR⁶ nebo XSS⁷. Framework obsahuje kvalitně zpracovanou dokumentaci, video návody ze strany vývojářů a dokonce i možnost online či telefonní konzultace s experty [7].

²CoC - Convention over Configuration

³DRY - Don't repeat yourself

⁴SASS - Syntactically Awesome Style Sheets

⁵CRUD - Create, Read, Update, Delete

⁶CSFR - Cross-site Request Forgery

⁷XSS - Cross-site Scripting

Stinnou stránku ztělesňuje především rigidita⁸ výchozích URL odkazů. Ačkoliv se jedná o robustní prostředek, jeho funkcionalita se oproti novějším frameworkům může v některých směrech zdát omezená [39].

Django

Se zaměřením na znovupoužitelnost a rychlost implementace vzniká v roce 2005 na základě vysokoúrovňového skriptovacího jazyka `Python` open-source framework, jenž se zaměřuje na tvorbu komplexních databázových webů [12]. Django nabízí standardní zabezpečení, širokou škálu funkcí a strukturovanost projektu v něm vytvořených [13]. Mezi netradiční možnosti patří spuštění FASTCGI serveru s vlastním administrativním rozhraním pro jeho správu [12]. Django staví na modelu MTV, který je s konvenčním modelem MVC takřka ekvivalentní [9]. Díky popularitě jeho a jazyka v něm napsaném, existuje řada návodů a dokumentace zastřešující základní principy a použitelnost nástroje.

Hodná povšimnutí jsou zejména sady existujících knihoven, jež poskytuje použitý jazyk `Python`, které lze využít i na řešení více komplexních úloh. Další přednost reprezentuje objektově relační mapování a šablonovací systém [12].

Mezi hlavními negativy si své místo najde pomalejší učící křivka a nevhodnost užití na menších projektech z důvodu objemnější výchozí struktury. Komplikaci může tvořit i monolitický charakter, jenž nemusí být kompatibilní s přístupem vývojáře a působit nepříjemnosti [29].

3.2 Front-end frameworky

V rámci analýzy front-end frameworků se seletovaly nástroje hojně skloňované v kontextu s kompatibilitou s MVC architekturou. Jedná se o populární prostředky s širokým polem působnosti, vydány softwarovými giganty jako Google či Meta. Poskytují vývojářům předpřipravené komponenty, jež mají proces implementace uživatelského rozhraní usnadnit a zrychlit.

Angular

Framework Angular, vydán v roce 2016 pod společností Google, tvoří jednu z esenciálních součástí vývojářského balíčku MEAN, jež je sestaven z iniciál nástrojů MongoDB, Express.js, Angular a Node.js [21]. Podpora jeho předchůdce AngularJS, napsaném v jazyce `Javascript`, byla ukončena koncem roku 2021 za přesunutí pozornosti a zdrojů na vývoj jeho novější `Typescript` verze [17].

Programovací jazyk `Typescript` rozšiřuje jazyk `Javascript` o možnosti jako statické typování či podpora modularity [19]. Jejich vztah se může připodobnit ke vztahu nízko úrovněových jazyků `C` a `C++`.

Jako největší přednost lze považovat možnost datové vazby mezi komponenty Model a View, která je v případě modifikace instantně změněna v obou současně. Pro znovupoužití slouží vložení závislosti na specifický modul, čímž je vytvářena

⁸Rigidita - tuhost, neohebnost

hierarchická struktura programu. Nástroj se dále velebí svým výkonem a jednoduchou testovatelností [33].

Kvůli své komplexnosti, velikosti a úsilí nutnému k vynaložení na naučení není optimální pro menší vývojářské týmy a projekty nevelkého charakteru [22].

React.js

Kvůli rostoucímu množství funkcí webového systému Facebook se jeho vývojáři rozhodli v jazyce `Javascript` sestavit vlastní knihovnu s názvem React za účelem maximalizace přehlednosti kódu, do níž celou sociální síť v roce 2011 začali převádět. Dva roky poté se knihovna stala open-source, ale kvůli skepticizmu komunity se popularity dočkala až s odstupem času [20]. React je typický pro své DOM⁹ rozhraní, jež reprezentuje celý dokument ve formě stromovité struktury [40].

Díky své robustní povaze doprovázené pravidelnými aktualizacemi, dokáže tento nástroj poskytnout široké spektrum funkcí [22]. Silná stránka spočívá především v recyklovatelnosti jednotlivých komponent, rozšíření JSX¹⁰ a v tzv. hoocích, které umožňují používat framework bez nutnosti tříd [40]. Při práci na komplexních nebo jednostránkových webech se jedná o jednu z nejpreferovanějších knihoven současnosti.

Komplikace nastávají v sebevzdělávací fázi, a to primárně ve složité syntaxi rozšíření JSX, obzvláště pro vývojáře bez zkušeností s jazykem `Javascript` [33]. Časté aktualizace mohou způsobit, že se brzy po vydání jeho podpůrné materiály jako dokumentace nebo návody stávají neaktuálními, a tak se vývojáři musejí často nové principy a funkce doučovat sami [40].

Vue.js

Knihovna Vue.js vznikla v roce 2014 rukou autora, jež se dříve podílel na vývoji frameworku Angular 3.2 společnosti Google [5]. Za ústřední myšlenkou stálo vytažení oblíbených částí nástroje a vytvoření jeho odlehčené verze [33]. Výsledek představuje výrazně menší a simplifikovanější balík funkcí, který si dokázal najít své oblíbence, obzvláště z řad vývojářů východní Asie [22]. Kuriózními jsou netradiční názvy jednotlivých verzí pojmenovávány známými japonskými anime či manga tituly [38].

Popularity se dočkal především kvůli svému rozhraní DOM a datovým vazbám, doprovázené flexibilním a pochopitelným charakterem [33]. Ocenit lze kvalitně zpracovanou dokumentaci a rozšíření nabízené vydavatelem [22]. Využívá se převážně k tvorbě jednostránkových webů nebo při práci na neokázalých projektech.

Framework se pro svoji flexibilitu může občas jevit nekonzistentní. Neoplývá přílišnou podporou uživatelů, neboť velkou sortu komunity tvoří obyvatelé Číny, kteří raději volí ke komunikaci svůj rodný jazyk [40]. Další její komplikace jsou způsobeny nezastřešením aplikace korporátní společnosti. Knihovna není vhodná pro vývoj komplexních produktů z důvodu neisté stability na aktualizovaných verzích prohlížečů a zpochybnitou bezpečností [22].

⁹DOM - Document Object Model

¹⁰JSX - Javascript Syntax Extension

3.3 Shrnutí

Vzhledem k aktuálnosti a popularitě webových stránek, existuje nástrojů na jejich vytvoření celá řada, a v rámci analýzy jich bylo možné pokrýt pouze zlomek. Omezení průzkumu může pocházet především z nepokrytí veškeré funkcionality jednotlivých nástrojů a jejich neotestování na reálných projektech.

Back-end frameworky

Framework	Výhody	Nevýhody
Ruby on Rails	syntax, konfigurace, jednostránkové aplikace	rychlost, nároky na výkon, dokumentace
CakePHP	bezpečnost, učící křivka, dokumentace	omezená funkcionality, rigidita URL
Django	funkcionality, dokumentace, strukturovanost	učící křivka, jednotvárnost

Vybrané aplikační frameworky spojuje zaměření na znovupoužitelnost a snaha o simplifikaci konfigurace a struktury projektu.

Front-end frameworky

Framework	Výhody	Nevýhody
Angular	závislosti, testovatelnost, podpora, výkon	učící křivka, mohutnost
React.js	robustnost, funkcionality podpora, strukturovanost	učící křivka, dokumentace
Vue.js	učící křivka, flexibilita, dokumentace	nestabilita, komunita bezpečnost

Mezi prezentačními frameworky se kromě znovupoužitelnosti nachází majoritní snaha o přívětivé zpracování datových vazeb.

Pro implementaci práce se využije k vytvoření serverové části nástroj **Django**, kvůli jeho široké funkcionalitě potřebné na vypracování netradičních požadavků, jako manipulace s mapovým podkladem či práce s polygony kamer. Dále z důvodu uživatelské přívětivosti jazyka **Python** a osobních preferencích.

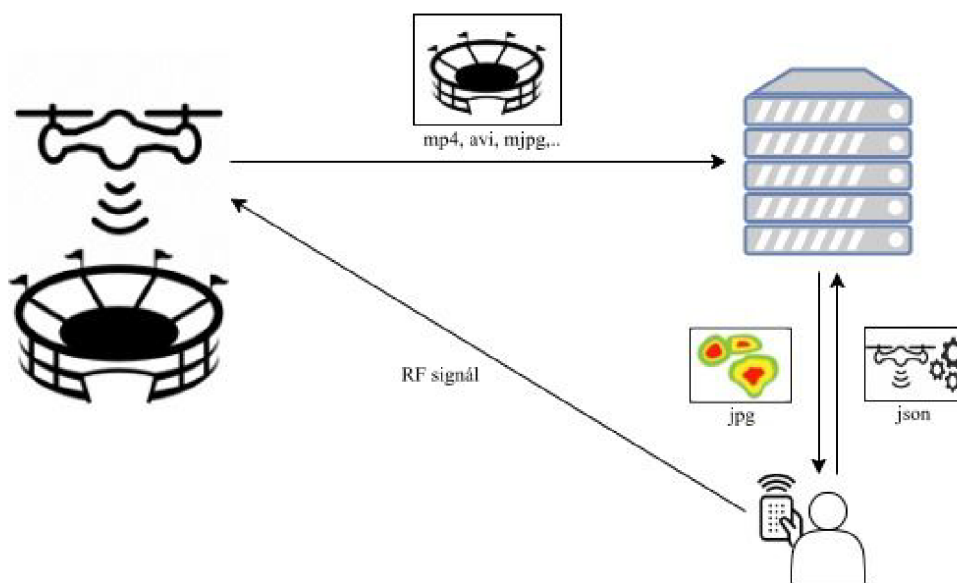
Klientskou stranu bude zastřešovat knihovna **React**. Tomuto robustnímu frameworku se dostává řada podpory ze strany vývojářů i komunity, díky které si udržuje stabilitu napříč aktualizovanými prohlížeči a dostává se mu dostatečné množství podpůrných materiálů. Upřednostněn oproti nástroji **Angular** je na základě větší popularity a méně komplexnější syntaxi.

Kapitola 4

Implementace

Síťová architektura výsledné aplikace pracuje na modelu klient-server, jež staví na principu odesílání požadavků uživatele na cílovou stanici a její následné zpracování s poskytnutím odpovědi ze strany serveru [27]. Kromě standartních žádostí o poskytnutí specifických webových stránek, tvoří značná většina komunikace asynchronní požadavky na přenos video dat a manipulaci s databázovými reprezentacemi dronů. To je dáno především single-page charakterem webové aplikace, která vyjímá autentizační části představuje veškerý svůj obsah v klientském prohlížeči na jediné webové adrese [2].

Ačkoliv v kompletním kontextu komunikace představuje server důležitý zdroj dat, jejich majoritní podíl nevytváří, nýbrž distribuuje 4.1. Tuto část, pocházející ze streamovacích odkazů dronů poskytnutých uživatelem, server nejen přeposílá, ale také využívá k vypracování heatmapy anomálií. Poslední pilíř komunikačního schématu tvoří uživatel, který musí kromě práce s aplikací také bezpilotní letadlo v dohledové vzdálenosti dle doplňku X o bezpilotních systémech ovládat [35]. To za účelem minimalizace rizika kolize s okolním prostředím a ohrožením tak jeho či dronu.



Obrázek 4.1: Schéma komunikace

Hierarchickou strukturu projektu tvoří dvě synteticky vytvořené aplikace¹. První s názvem **api** ukrývá veškerou implementaci funkcí webového rozhraní a komunikaci s databází, čímž představuje serverovou část. Ve druhé označené **frontend** lze naopak nalézt veškeré prezentační prvky a tudíž ji separovat jako klientskou část.

- **api/** - datové modely, obsluha požadavků, URL požadavků a jejich mapování
- **frontend/** - prezentační komponenty, kaskádové styly, obrázky
- **project/** - konfigurace projektu
- **videos/** - video záznamy dronů

Celý implementační proces se odehrál ve vývojovém prostředí **Visual Studio Code** za pomoci FastCGI serveru zprostředkovatelném back-end frameworkem **Django**.

4.1 Serverová část

Hostitelský program, implementován ve skriptovacím jazyce **Python** za využití aplikačního frameworku **Django**, poskytuje prostřednictvím MTV² modelu služby podobně jako konveční model MVC. Nejprve dojde k vytvoření požadavku u klienta na Template vrstvě a jeho odeslání. Po jeho obdržení se na straně serveru požadavku zmocní View vrstva, jež namapuje URL žádosti na příslušnou aplikační logiku a pomocí získaných perzistentních dat z nejhlubší Model vrstvy požadavek zpracuje. Poté je View vrstvou vygenerována adekvátní odpověď, přeposlána Template vrstvě a následkem toho vhodně dynamicky modifikována prezentační HTML struktura šablony klienta [9]. Oproti architektuře MVC se redukuje míra úsilí nutného vynaložit na implementaci alter ega její Controller komponenty, View vrstvy, kterou framework řeší za vás.

Jednotlivé **Django** aplikace¹ se skládají z několika dílčích upravitelných souborů reprezentující funkcionalitu serveru. Mohou být využívány ve více projektech zároveň, jejich selekce je podmíněna importem v projektovém konfiguračním souboru **settings.py**.

- **__init__.py** - prázdný soubor deklarující aplikaci, dále se neupravuje
- **admin.py** - správa superuživatele a jeho rozhraní poskytnuté frameworkem, jež poskytuje možnost kdykoliv jednoduše modifikovat objekty jednotlivých modelů
- **apps.py** - konfigurace aplikace, měněn jen vzácně
- **models.py** - definice datových modelů tvořící kostru databáze, pro propagaci změn, jako přidání atributu či vytvoření nového modelu, do již zavedené struktury je nezbytné provést migraci prostřednictvím série jednoduchých příkazů
- **tests.py** - soubor, oplývající vytvořitelnými testy specifických celků aplikace
- **urls.py** - list URL s jejich příslušnými Views, využíváno k URL mapování
- **views.py** - aplikační logika jednotlivých Views, obsahuje přijetí dat, jejich zpracování, komunikaci s Model vrstvou a vygenerování odpovědi tázanému zařízení

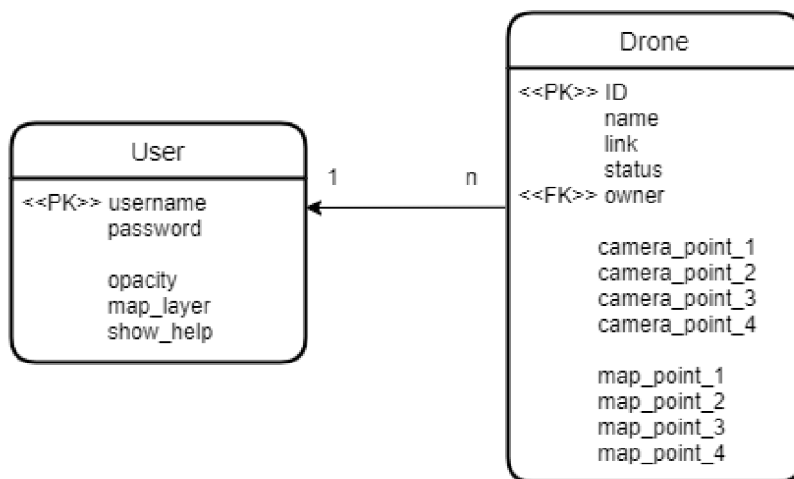
¹aplikace - elementární jednotka projektu vytvořená frameworkem **Django**

²MTV - Model, Template, View

Aplikace **api** je obohacena o soubor **serializers.py**, jehož obsah se běžně vyskytuje ve **views.py**, ale za účelem strukturovanosti byl separován. Nachází se v něm šablony na serializaci dat klientu odeslaných a deserializaci dat serverem obdržených.

Modely

Řešení využívá dvou základních modelů, jeden představující samotné uživatele a druhý bezpilotní letadla jimi využívaná. Hesla uživatelů se z důvodu ochrany osobních údajů [34] nachází v databázi v zahashované podobě a při autentizaci dochází k tradičnímu procesu zahashování nově příchozí hodnoty hesla stejným algoritmem a porovnání se zašifrovanou hodnotou v databázi.



Obrázek 4.2: ER diagram

Spolu s jménem a heslem obohacuje uživatelskou entitu trojce vlastností, které své uplatnění přináší v individuálním nastavení aplikace. Oproti uložení dat ve formě cookies v prohlížeči se perzistence zajišťuje prostřednictvím databáze z důvodu zvýšené frekvence obměny využívaných zařízení bezpečnostních složek.

Reprezentace dronů kromě očekávaných atributů, jako název či odkaz na stream, nesou informace o dvou sadách 4 bodů využívaných k projektivní transformaci. První sada stojí za relativní pozicí hraničních bodů příchozích snímků procentuálně vyjádřených. Obor hodnot se tedy pro obě dimenze rozpíná mezi 0, kdy se bod nachází na jedné ze dvou počátečních hran kamery, a 100, neboli na hranách protějších. Druhou čtveřici představují transformační body ležící na mapě ve formátu zeměpisné šířky a výšky. Kompletní proces transformace zachycen v kapitole 4.2.

Video záznamy

Na zpracování video stopy poskytla pomocnou ruku open-source knihovna **OpenCV**³, jejíž funkce zastřešují téměř celý akt počínající zachycením dílčích snímků z toku video dat až po jeho přeformátování do JPEG formátu, který je následně odeslán klientovi. Souborový formát JPEG využívá principů ztrátové komprese, není vhodný

³OpenCV - <http://opencv.org>

pro zachování vysokých barevných rozlišení s ostrým kontrastem, ale pro účely vyobrazení rozsáhlých krajin, jež jsou od cílové skupiny uživatelů očekávány, se jeví užitečným [11].

Pro příjem video dat odešle klient požadavek s názvem svého zařízení, na který server reaguje vyhledáním streamovacího odkazu a přečtením jednotlivých snímků z jeho datového toku na nově vytvořeném vlákně. Kromě něj utvoří další vlákno na zaznamenávání snímků a zapisování videa do paměti serveru pro případnou zpětnou rekonstrukci stopy. Ten se uloží ve složce **videos/** do podsložky s unikátním přihlašovacím jménem uživatele pod identifikátorem zařízení.

Pokud byla vznešena žádost na přenos specifické části již proběhlého streamu, využije tázaný počítač doposud zapsané stopy a simuluje tok dat synteticky. Nejprve dojde ke zjištění celkového počtu snímků přetočením záznamu, neboť metadata, kvůli probíhajícímu zápisu videa do nevolatilní paměti, tuto informaci ještě neobsahují. Následně se provede skok a posun videa na žádaný počáteční bod streamu díky dvěma řídicími klientem zaslánými argumenty. První je zprostředkován pohybem ukazatele po časové ose videa, který stojí za procentuální reprezentací aktuální doby záznamu, a druhý se poskytne zmáčknutím levého či pravého posuvného tlačítka, jež dekrementuje nebo inkrementuje svůj real-time čítač o časový horizont 5 sekund 4.3. Jednotlivé čítače ožívají při obdržení odpovědi o započítání odebrání toku video dat ze strany serveru a jejich hodnoty se na základě jejich interakce průběžně upravují. Odeslané informace se tedy skládají z pozice videa na intervalu od 0 do 1 a celočíselném posunu v sekundách.



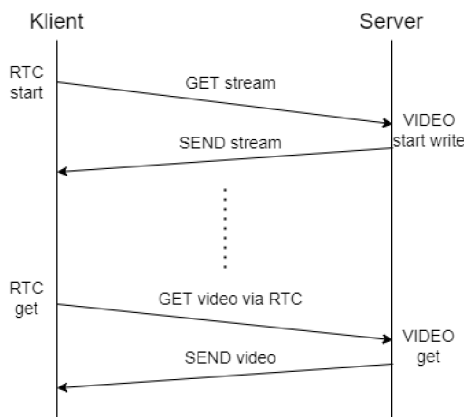
Obrázek 4.3: Přehrávač živého vysílání s prvky videa na vyžádání

Výkyvové nuance⁴ zpoždění, způsobené přenosovou dobou mezi časem odeslání požadavku a přijetím odpovědi je nutné eliminovat z důvodu aktuálnosti vyžádané video stopy. Ty se minimalizují o rozdíl délky počáteční a aktuální reakce komunikace. Časové kontinuum začíná vytvoření umělého RTC⁵ časovače u tázajícího a následuje odstartováním zápisu na hostitelském zařízení 4.4. Při další komunikaci je poté pož-

⁴nuance - jemný rozdíl, odlišnost

⁵RTC- Real-time clock

dění exterminováno využitím zvětšené hodnoty časovače oproti délce vzniklé video stopy na serveru.



Obrázek 4.4: Časové schéma požadavku na přenos zapisovaného videa

Video záznamy jsou dostupné pouze po dobu běhu aplikace a při dalším spuštění dojde k jejich přepsání.

Požadavky na server

Při výběru vhodného počítače určeného k převzetí břímě hostingu je důležité brát ohled na řadu faktorů a výkonových požadavků aplikace. Díky absenci perzistentních dat v souborovém systému se jeví možné volit decentralizovanou správu paměti. Při výběru velikosti paměťového prostoru hraje markantní roli maximální počet aktivních zařízení a průměrná doba jejich působení, neboť pořízené video záznamy představují primární zátěž uložště.

Problematika výkonu může oproti paměťovým nárokům působit pouze znepokojení marginální, ale stále nezanedbatelné. Server dokáže konstantně přijímat, zpracovávat a odesílat hned několik toků video dat na klienta, což při větším množství odběratelů může procesor značně vytížit. Maximální počet aktivních zařízení tudíž znovu reprezentuje klíčový faktor.

Zavrženíhodná není ani lokalita serverovny pro aktuálnost přijetých dat z důvodu nutnosti adekvátní včasné reakce bezpečnostních složek. Kromě toho by hostitelská oblast měla poskytovat možnost stabilního připojení a dostatečně velikou šířku pásma pro obdržení i odeslání dat. Za kritérii selekce lokality tedy stojí centralizace skupiny uživatelů a dostupnost internetových služeb.

Serverové požadavky na 10 aktivních zařízení

Vlastnost serveru	Minimální požadavky	Optimální požadavky
Velikost paměti	87.5 GB	100 GB
Procesor	Intel Core i5-10600K	Intel Core i7-10700KF
Rychlost uploadu	40.95 Mbs	77.6 Mbs

Parametry v tabulce slouží pouze orientačně. Pro její vyhotovení byly využity videa s Full HD rozlišením o 25 FPS a průměrnou dobou aktivity 150min.

4.2 Klientská část

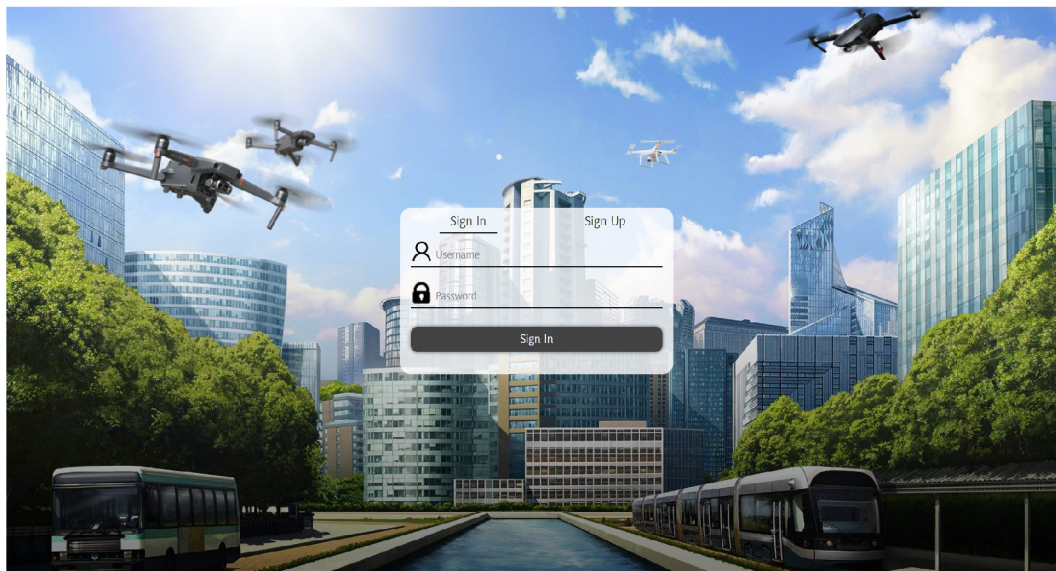
Stěžejní prostředek aplikace v podobě uživatelského rozhraní drží pohromadě frontend knihovna **React**, jež nabízí své základní stavební kameny, komponenty [3.2](#). Je zvykem každý projekt sestrojený tímto frameworkem započít mateřskou komponentou **App**, která v aktuálním řešení slouží jako směrovač webových adres. Tento prvotní domovský kořen dále větví veškeré části kódu do rozsáhlé stromovité struktury s aspekty dědičnosti a možností dynamického generování specifických uzlů. Soubory s tímto charakterem se nachází v podsložce **src/**.

- **App.js** - Rozcestník, obsahuje komponentu **App**
- **home_page.js** - majoritní funkcionalita, jádro aplikace
- **login_page.js** - autentizace, registrace

Vyhotovělá kostra je náležitě ozvláštněna kaskádovými styly a obrázky s lokalizací v podsložce **static/**. Úprava a konstrukce vybraných grafických prvků proběhla ve volně dostupném rastrovém editoru **Krita**.

Autentizace

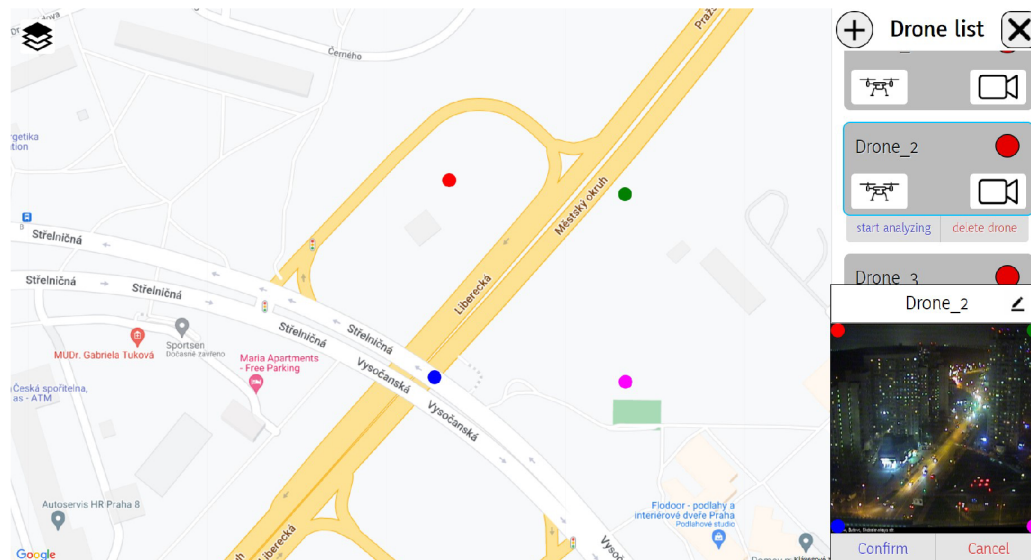
Při prvním setkání s webovým rozhraním jsme konfrontováni přihlašovacím oknem. Koncept uživatelů se jeví nezbytným kvůli separaci personálního obsahu a omezení přístupu nežádoucích osob, ačkoliv dostupnost registrace není prozatím selektivně omezena. Mezi přihlášením a registrací lze libovolně přepínat v horní liště podokna nad vstupními textovými poli [4.5](#).



Obrázek 4.5: Autentizační okno

Autentizace, z důvodu rychlejšího přístupu při testování, klade nárok na sílu hesla minoritně a to v podobě alespoň 7 zadaných znaků. Při nasazení do provozu se kvůli rapidním výpočetním výkonům moderních počítačů a novodobým přenosovým rychlostem jeví nezbytné tyto požadavky náležitě zintenzivnět. Po vložení již zmíněného

mění získáváním odpovědí na požadavky typu vytvoření či vymazání dronu. Po zvolení bezpilotní entity se prostřednictvím asynchronní komunikace odešle požadavek na opatření hodnot jeho bodů nacházející se vně kamery 4.7. Body mohou být v rámci boxu snímacího přístroje následně přemístěny tažením a položením, stejně jako jejich korespondující mapové dvojníci. Pro zvýšení přesnosti ukotvení pozic bodů není v tomto režimu transformované video v mapovém podkladu viditelné, jelikož by mohlo zastínit relevantní geografické objekty.



Obrázek 4.7: Nastavení pozice dronu

Nové nastavení, včetně modifikovatelného názvu zařízení, lze v případě satisfakce tlačítkem **Confirm** potvrdit, jež zasláním požadavku zapříčiní synchronizaci s daty v databázi. Pokud se ovšem uživatel dopustil omylu a nevědomě upravil již funkční celek, je možné celou akci stornovat stisknutím sousední buňky **Cancel**, která díky předuloženým hodnotám přeskupí upravitelné objekty do původní stavu.

Chceme-li pro specifický stream využít služeb generování heatmapy anomálií, nabízí se u selektované entity možnost započítí či ukončení analýzy skrze jedno ze dvou zpoza vysunutých zmáčknutelných prvků. Zařízení, provádějící analýzu, se změnou datového toku na snímky teplotní mapy, rozsvítí zeleně. Aplikace je formátově připravena tato data odesílat, ale prozatím tímto detekčním softwarem nedisponuje.

Druhé tlačítko zašle serveru žádost o perzistentní odstranění dronu. Nejprve se však rozhraní dotáže uživatele, jestli s akcí souhlasí a nedošlo pouze k nechtěné intervenci. Omezení užívání ostatních funkcí během klíčových událostí jako je tato, řeší CSS atribut **z-index**, jehož zvýšená hodnota zapříčiní zakrytí jiných elementů a bývá v aplikaci hutně uplatňován.

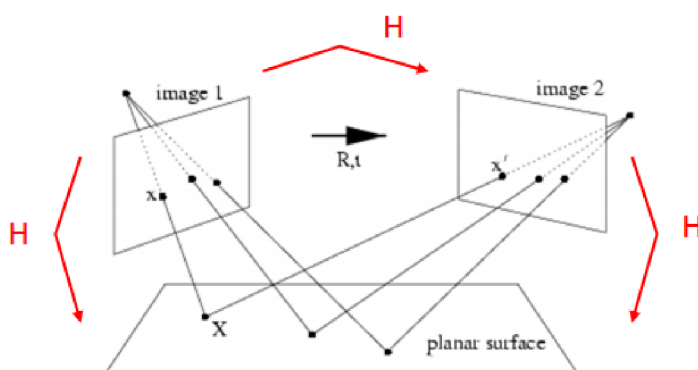
Veškeré animace v tomto či jiném segmentu pracují na principu změny umístění prvku z neviditelné do viditelné sféry, nejčastěji vložení na souřadnice přesahující velikost okna.

Za poslední nejvýraznější implementovanou vlastností seznamu bezpilotních zařízení stojí scrollovatelnost, která za využití výšky jednotlivých zařízení, obrazovky,

nadpisů a podoken kalkuluje rozměr listu, a při překrytí zformuje scrollovatelné prostředí 4.7.

Projektivní transformace

Autentická vizualizace video dat představuje klíčovou roli v chodu celé aplikace. Jelikož zachycení mapového podkladu a obraz kamery dronu reprezentují různé 2D interpretace stejného 3D prostoru, nelze jejich dvou dimenzionální projekce jednoduše přemístit z jedné do druhé. Snímek musí být adekvátně transformován, aby v kontextu druhé projekční roviny odpovídal jeho vyobrazení reality. Řešení tedy najdeme pouze v tradičním afinním zobrazení, založeném na zachování kolinearit⁶ a dělicího poměru⁷ mezi vzorem a obrazem, ale i v perspektivním zobrazení, typickému pro své vedení projekčních paprsků z jediného bodu. Kombinace těchto zobrazení se nazývá projektivní zobrazení neboli kolineace [1]. Pokud projektivní zobrazení stejného prostoru promítneme na sebe, vznikne **projektivní transformace**, také známá jako **homografie** 4.8 [26].



Obrázek 4.8: Nalezení homografie dvou snímků stejného prostoru (převzato z [4])

Účelem homografie je naleznout 3x3 transformační matici, díky které pak lze snímky ve formě elementů prostřednictvím CSS atributu `matrix3d` náležitě přeměnit. Funkci pro projektivní transformaci zprostředkovává knihovna `Homography.js`⁸ na jejíž vstupu stojí dvě skupiny 4 bodů, jedna pro každý obraz. Hodnoty obou se pohybují v procentuálním vyjádření a ve 4. kvadrantu kartézské soustavy souřadnic s počátkem v levém horním rohu a invertovanou hodnotou osy Y, jak je tomu u práce s maticí pixelů na obrazovce běžné.

První čtveřice lokalizovaná v záběru kamery začíná ve výchozím nastavením červeným bodem na souřadnicích $[0, 0]$ a hraničí protilehlým růžovým bodem v $[1, 1]$ 4.7. Soustava vznikla pomyslnou translací prvku do nulových koordinací okna za odečtení patřičné šířky a výšky. Vydělením vnitřních bodů s rozměry prvku si nakonec opatříme jejich procentuální reprezentaci, jež se uplatňuje v procesu hledání homografie.

S polem působnosti v kartografickém prostředí se druhá sada musí nejprve převést Mercatorovým zobrazením, jež eliminuje efekt zakřivení Země a exterminuje zkreslení

⁶kolinearita - zachovává po zobrazení rovnoběžnost

⁷dělicí poměr - po zobrazení udržuje stejný poměr vzdálenosti bodů na přímce

⁸Homography.js - <https://npmirror.com/package/homography>

projevující se s vzrůstající zeměpisnou výškou. Konverze využívá knihovných funkcí volně dostupného rozhraní `mercator-projection`⁹, které ze zeměpisných souřadnic vytvoří souřadnice zachovávající linearitu. Dále následuje stejný proces translace, tentokrát v kontextu mapového pokladu, jehož počátek se rozpíná ze specifické geografické lokace ukotvené v klientském levém horním rohu obrazovky. Výpočet koordinací této lokality probíhá za využití hodnoty přiblížení `zoom`, konvertovaných souřadnic středu mapy `center` a šířky okna `screen_width` 4.1. Protější jihovýchodní bod je kalkulován podobně a to záměnou subtrahce za adici před zlomkovým segmentem. Je-li provedena operace divize šířkou obrazovky na interní body mapy, získáváme cílovou relativní soustavu souřadnic.

$$\begin{aligned} x &= center.x - \frac{\frac{1}{2} \times screen_width}{2^{zoom}} \\ y &= center.y - \frac{\frac{1}{2} \times screen_width}{2^{zoom}} \end{aligned} \tag{4.1}$$

Nyní vlastněme 2 sady relativních souřadnic, jež lze použít k zformování transformační matice. Pro každé zařízení probíhá její sestavení vždy po přihlášení uživatele, upravení pozice bodů či vytvoření nového dronu. Prvek před přeměnou se nejprve zobrazí ve čtvercové podobě o délce šířky okna (proto v 4.1 y-souřadnice pracuje s `screen_width` místo `screen_height`) s počátkem opět v levém horním rohu obrazovky. Korektní vizualizace nakonec dosáhneme uvedením hodnoty CSS atributu `transform-origin` na left top a dynamickou aplikací atributu `matrix3d` vyžadující na vstupu matici 4x4 u které se zbavíme z-tové dimenze a patřičně upravíme specifické buňky, kvůli zpětnému převodu do absolutních jednotek 4.2.

$$M = \begin{vmatrix} m[0] & m[3] & 0 & m[6]/w \\ m[1] & m[4] & 0 & m[7]/w \\ 0 & 0 & 1 & 0 \\ m[2] \times w & m[5] \times w & 0 & 1 \end{vmatrix}$$

m = transformační matice w = šířka okna

Samotná funkce evokující transformaci video dat se volá s každou změnou měřítka mapy nebo úpravou rozměrů okna.

Nabídka dronu

Kliknutím na transformovaný element, případně stisknutím symbolu bezpilotního letadla v listu dronů, přesměruje uživatele do hlavní nabídky dronu a adekvátně upraví aktuální střed kartografické struktury na pozadí, aby se pozorovaná oblast nacházela ve volném prostoru po levé straně 4.9 a v případě aktivní analýzy bylo možné situaci sledovat paralelně s neanalyzovanou video stopou.

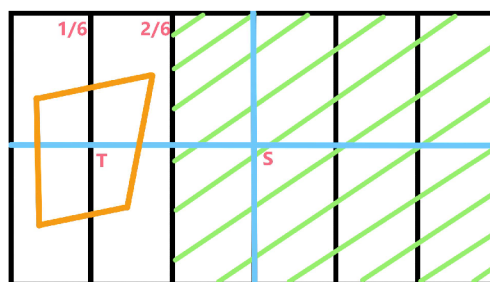
Implementace proběhla pomyslným řezem okna na 6 rovných dílů, z kterých dvě třetiny pokrývají menu. S nejdelší vzdáleností dvou krajních x hodnot polygonu se

⁹mercator-projection - <https://www.npmjs.com/package/mercator-projection>



Obrázek 4.9: Hlavní nabídka dronu

nejdříve zajistí největší potřebné přiblížení, aby šíře objektu nepřesahovala $\frac{1}{3}$ šířky obrazovky. S vypočteným těžištěm ze 4 dostupných koordinací a znalostí přiblížení lze zpětnou konverzí do zeměpisných souřadnic určit cílovou destinaci směřování 4.10.



Obrázek 4.10: Schéma separace nabídky dronu

Na limitace daného algoritmu narazíme při nekonvenčních polygonech a u podlouhlých výsledků transformací, které nemusejí být vyobrazeny nejelegantněji, avšak nejsou v rámci případů užití příliš běžné.

Hlavní nabídka poskytuje mírně zkrácenou video stopu, škálovanou z estetických důvodů, tlačítko přepnutí hlavní kamery a přesměrování na úpravu pozice dronu. Redundantní směrovací prvky jako tyto nabízejí uživateli vybrat z více způsobů využívání aplikace dle jeho preferencí a aktuálních potřeb.

Kamera dronu

Za posledním funkčním celkem stojí přehrávač nezkrácených záběrů s rozšířením o vlastnosti videa na vyžádání, více o principu zpracování konkrétních stop v podkapitole 4.1. Rozhraní oplývá konvenčními ovládacími prvky jako tlačítka posunutí, jezdcem a časovou osou jím využívajícím 4.12. Jednotlivá videa si zachovávají hodnoty jezdců a navzdory jejich inaktivitě, způsobenou užíváním celku jiným zařízením,

se současně pohybují, což má za následek možnost sledování vícero starších záznamů v libovolném čase.



Obrázek 4.11: Hlavní kamera dronu

Pozici jezdce definuje čítač, spuštěný od začátku odebrání video dat ze serveru, a procentuální reprezentace aktuální pozice upravitelná uživatelem. Ve výchozím nastavení přijímá přehrávač živé záběry s hodnotou jezdce 1. Pohnutím jezdce lze jeho chování abstrahovat do základního vztahu 4.2.

$$P = \frac{t}{t_s} \quad (4.2)$$

Hodnota P představuje relativní pozici jezdce na ose, t reprezentace času, na kterém se jezdec nachází a t_s délku času doposud přijatého streamu. Hodnoty t a t_s se průběžně inkrementují prostřednictvím syntakticky vytvořeného RTC modulu a t je upravován posuny v čase, které vyvolává stisknutí symbolů přetočení.

Mapové vrstvy

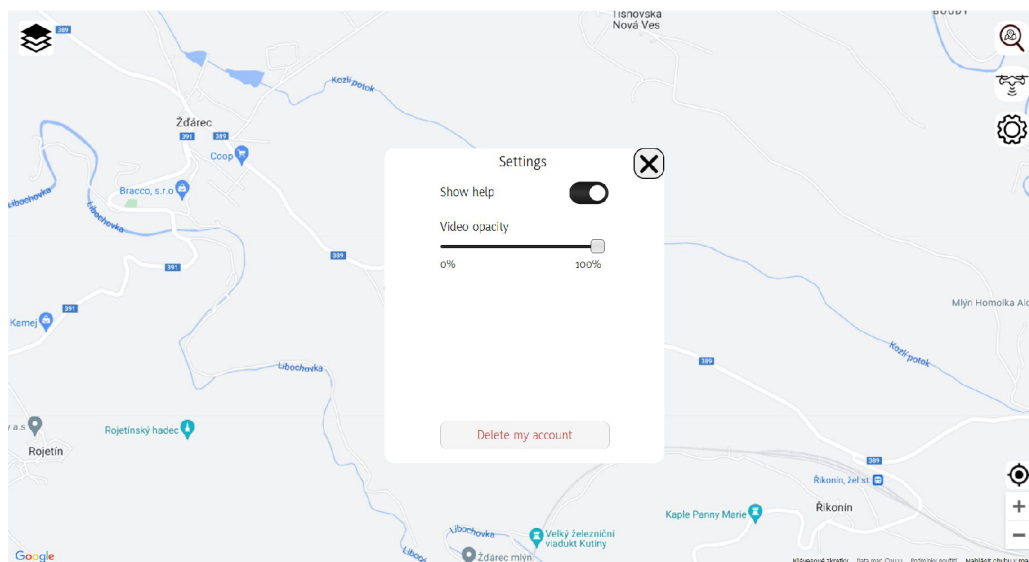
Za účelem poskytnutí rozmanitých kombinací sledování videí a heatmap, jež může uživatel chtít uvést do chodu, vznikla možnost přepnutí mapových vrstev. Po stisknutí konvenční ikony na protějším konci hlavní nabídky se objeví dvojice typů map se specifickou funkcionalitou, které lze vybráním aktivovat.

- **Klasická** - Vyobrazuje pouze video data kamer
- **Heatmapa** - Vyobrazuje video data kamer a u dronů s aktivní analýzou jejich generované heatmapy

Díky spolupráci mapových vrstev s aktivními analýzami se může jednoduše přepínat mezi zobrazovacími režimy a dosáhnout žádoucího stavu monitorování. Vrstva se po selekci ukládá nevolatilně.

Nastavení

Pro minimální optimalizaci a zvýšení komfortu lze upravit pár dílčích prvků aplikace za využití sekce nastavení. Ty jsou stejně jako mapové vrstvy perzistentně zachovány v databázi, čímž se stávají přenositelné mezi zařízeními. Nastavení obsahuje funkce vypnutí nápovědy a změnu průhlednosti videí na mapovém podkladu.



Obrázek 4.12: Nastavení s výchozími hodnotami

Za poslední možností stojí smazání účtu, které se před trvalým odstranění uživatele s jeho registrovanými zařízeními dodatečně dotáže, jestli s danou nevratnou akcí souhlasí. Při pozitivní odpovědi pak kromě vymazání účtu dojde k přesměrování na domovskou stránku s autentizací.

Požadavky na klienta

V rámci výběru optimálního počítače využívaného pro běh aplikace nesmí být opomenuta řada minimálních kritérií a dostupných služeb. Pro uživatele tabletových zařízení se silně doporučuje z poměrových charakterů videí užívat režim na šířku. Pokud máte k dispozici zařízení několik, největší prioritou by měla být vložena jedinci s nejdelší úhlopříčkou.

Další klíčovou roli hraje internetové připojení. Jeho stabilita závisí především na míře akutnosti přijímaných dat, ale celkově se nedoporučuje software využívat v příliš nepokrytých oblastech jako přírodní parcích či lesích. Rychlost připojení se opírá o množství používaných zařízení s očekávaným počtem streamů. Stahovací hodnoty lze poměrově připodobnit k rychlostem nahrávacím v 4.1. Pro mobilní sítě by mělo být maximální množství přijatých dat ideálně neomezené.

Výkonové požadavky aplikace se odvíjí od používaného prohlížeče, ten je nutný pro optimální zabezpečení a zachovanou funkcionalitu udržovat aktualizovaný.

4.3 Testování aplikace

Dlouhodobým užíváním internetu a používáním mimořádně kvalitních služeb na něm se nacházejících si obyvatelé tohoto virtuálního světa vybudovali vysoké požadavky na spolehlivost a výkon aplikací. Kvůli širokému spektru dostupných softwarových produktů může již minoritní nedostatek přeměrovat uživatele na produkt jiný, zvláště u řešení nekomerčních. V rámci životního cyklu vývoje se tedy jeví nezbytné co nejvíce těchto chyb odstranit, ať už procesem verifikace, kdy dochází ke kontrole, že implementace odpovídá specifikacím a návrhu, nebo procesem validace, při které se ověří neodchýlení od skutečných tužeb a přání zákazníka [30].

Jelikož byla aplikace pravidelně podrobována testům funkčních celků během implementační fáze a proces validace by se z důvodu absence dostupných prostředků a již provedených návrhových testů nemusel jevit příliš efektivně, zaměřuje se aktuální testování pouze na verifikaci. Oproti testování v kapitole 2.4 se navíc celý proces stal více fluidním, aby náležitě rozšířil spektrum potenciálních uživatelských vstupů.

Testovací protokol

Testování probíhalo na úrovni jednotlivců s délkou 10-15 minut, zahrnujíc všechny aspekty procesu vyjímaje vyhledání místa.

Metodika

1. Nalezení optimálního místa s dostatečným internetovým připojením a bez externích rušivých elementů
2. Seznámení subjektu s účelem testování, průběhem a jeho úkolem.
3. Splnění úkolu v aplikaci a volný pohyb jeho funkcemi s obdržáním zpětné vazby, po němž se test ukončí.

Úkoly

1. Zaregistrujte se a přihlaste se do aplikace.
2. Vložte do aplikace živý přenos dvou dronů. K dispozici máte v otevřeném textovém souboru streamovací odkazy na živé záznamy jejich kamer.
3. Zobrazte si aktuální dění na kameře prvního dronu a přetočte pořizovaný záznam na začátek vysílání.

Výsledky

Testem byli podrobena 4 účastníci ve věkové skupině 21-22, jež dosáhli středoškolského vzdělání. Vybírání byli především jednotlivci se zaměřením mimo technické obory. Jako výsledek jejich práce s aplikací nevznikala sada měřitelných hodnot, nýbrž informace o existenci konkrétních nedostatků a chyb. Testování odhalilo celkové množství čtyř chyb, dvě závažnějšího typu a dvě neinvasivního charakteru. Všechny závady byly před každou následující iterací testování opraveny.

Kapitola 5

Závěr

Za ústřední myšlenkou této práce stálo vyhotovení uživatelského rozhraní, jež by adekvátně vizualizovalo davové anomálie na video datech z dronu. Její výsledný produkt je připraven tyto data přijímat a v kontextu specifické lokality vyobrazit s obohacením uživatele o další řadu monitorovacích funkcí.

Proces vypracování řešení začal specifikací klientských požadavků a prostudováním literatury tvorby návrhů. Následně došlo iterativní metodou k jeho vytvoření s přihlédnutím k existujícím řešením v dané oblasti. Navazovalo sestavení makety prototypu k otestování návrhu, jejíž výsledky byly využity na odstranění slabých míst návrhu. Po rešerši moderních technologií na tvorbu webových aplikací a výběru některých z nich proběhla implementační fáze, která celé řešení přivedla k životu. To se nakonec opět otestovalo a modifikovalo za pomoci odhalených nedostatků. Výsledná aplikace reprezentuje softwarový produkt této práce.

Řešení se dále může opírat cestou optimalizace z důvodu aktuálně vysokých nároků na klientské zařízení i server. Kromě něj lze rozšířit sekci nastavení a vylepšit vytvořený přehrávač videí.

Literatura

- [1] Projektivní geometrie I – Pravý úhel pohledu. [online]. Matematický korespondenční seminář. 2019. s. 21.
- [2] AMANDA K.. What Is A Dynamic Web Page? [online]. Packet labs. 2021, [cit. 2022-04-12].
- [3] BAŽOUT, D. *Detekce anomálií v chování davu ve videodatech z dronu*. Brno, CZ, 2021. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií.
- [4] BRADSKI G., KAEHLER A.. *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, 2008. s. 163.
- [5] BREWSTER, C. *What Is Vue.js? The Pros and Cons of Vue.js in 2022* [online]. Trio [cit. 2022-04-12].
- [6] CAKE SOFTWARE FOUNDATION. CakePHP. [online]. [cit. 2022-04-13]. Dostupné z: <https://book.cakephp.org/4/en/plugins.html>.
- [7] CAKE SOFTWARE FOUNDATION. *CakePHP* [online]. [cit. 2021-12-28]. Dostupné z: <https://cakephp.org>.
- [8] CAKE SOFTWARE FOUNDATION. *Database Access ORM* [online]. [cit. 2022-04-12]. Dostupné z: <https://book.cakephp.org/4/en/orm.html>.
- [9] CORREIA, R. a ADACHI, E. Detecting Design Violations in Django-Based Web Applications. Association for Computing Machinery. 2019. s. 34.
- [10] CROZIER W. R.. *The meanings of colour: preferences among hues*. Pigment Resin Technology, 1999. s. 8.
- [11] DEWIA D. A. I. C., WIDYANTARA I. M. O.. Usage analysis of SVD, DWT and JPEG compression methods for image compression. Jurnal Ilmu Komputer. 2021. s. 100.
- [12] DJANGO SOFTWARE FOUNDATION. *Django* [online]. [cit. 2022-04-12]. Dostupné z: <https://docs.djangoproject.com/en/dev/faq/general/#what-does-django-mean-and-how-do-you-pronounce-it>.
- [13] DJANGO SOFTWARE FOUNDATION. *Django* [online]. [cit. 2021-12-28]. Dostupné z: <https://www.djangoproject.com>.

- [14] DRONEDEPLOY. [online]. [cit. 2021-12-21]. Dostupné z: <https://www.dronedeploy.com>.
- [15] FLANAGAN, D. a MATSUMOTO, Y. *The Ruby Programming Language*. O'Reilly Media, 2008. s. 2.
- [16] FLYTBASE, INC. [online]. [cit. 2021-12-20]. Dostupné z: <https://docs.flytnow.com>.
- [17] GOOGLE. *Version Support Status* [online]. [cit. 2022-04-12]. Dostupné z: <https://docs.angularjs.org/misc/version-support-status>.
- [18] HANSSON, D. H. *Rails 1.0: Party like it's one oh oh!* [online]. 2005.
- [19] HARTMAN, J. TypeScript Vs JavaScript: What is the Difference? [online]. Guru99. 2022, [cit. 2022-04-13].
- [20] HÁMORI, F. The History of React.js on a Timeline. [online]. RisingStack. 2018, [cit. 2022-04-12].
- [21] IBM CLOUD EDUCATION. *MEAN Stack* [online]. [cit. 2022-04-12]. Dostupné z: <https://www.ibm.com/cloud/learn/mean-stack-explained>.
- [22] KARCZEWSKI, D. *What Are The Best Frontend Frameworks To Use In 2021?* [online]. Ideamotive, 2021.
- [23] KIRAN B. R., THOMAS D. P., PARAKKAL R.. *An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos*. Journal of Imaging. Multidisciplinary Digital Publishing Institute, 2018. sv. 4, č. 2, s. 36.
- [24] KOŘOUSKOVÁ, B. Proč a kdy pro vývoj webových aplikací zvolit Ruby on Rails? [online]. Rascasone. 2021.
- [25] KRUG, S. *Don't make me think*. New Riders Press, 2000. s. 97, s. 46, s. 39.
- [26] LÁVIČKA, M. *KMA/G2 GEOMETRIE 2*. Západočeská univerzita v Plzni, 2006. s. 67.
- [27] MATOUŠEK, P. *Síťové služby a jejich architektura*. Publishing house of Brno University of Technology VUT IUM, 2014. s. 49.
- [28] MINISTERSTVO VNITRA. *Vyhláška č. 393/2006 Sb., Příloha č. 1*. 2006. §7, Oddíl III, Kód XIX, VII, V.
- [29] NADER, Y. What is Django? Advantages and Disadvantages. [online]. Hackr.io. 2022, [cit. 2022-04-12].
- [30] NAIK, K. a TRIPATHY, P. *Software Testing and Quality Assurance: Theory and Practice*. Wiley Publishing, 2018. s. 8.
- [31] NJENGA, A. *10 Popular PHP frameworks for web developers to consider in 2021* [online]. Raygun, 2021 [cit. 2021-12-29].

- [32] PATEL, J. *20 Best Ruby on Rails Gems to be Used in Web Development* [online]. Monocubed, 2021 [cit. 2022-04-14]. Dostupné z: <https://www.monocubed.com/ruby-on-rails-gems>.
- [33] PATEL, J. *List of 10 Best Front end Frameworks to Use For Web Development* [online]. Monocubed, 2021 [cit. 2022-04-13]. Dostupné z: <https://www.monocubed.com/best-front-end-frameworks>.
- [34] RADA EVROPSKÉ UNIE. *Nariženi Evropského parlamentu a Rady (EU) 2016/679*. předpis L 119/1, článek 5 (1-f).
- [35] RADA EVROPSKÉ UNIE. *Nariženi Evropského parlamentu a Rady (EU) 2018/1139*. Hlava 3, předpis L 2, doplněk X.
- [36] SANTAMARINA-CAMPOS V., SEGARRA-OÑA M.. *Drones and the creative industry*. Springer International Publishing AG part of Springer Nature, 2019. s. 117.
- [37] SECURITY EYE SOFTWARE. [online]. [cit. 2021-12-20]. Dostupné z: <https://www.security-eye-software.com>.
- [38] SHARMA, Y. *VueJs Version History* [online]. Tutorials Link, 2021 [cit. 2022-04-12].
- [39] SONI, D. *CakePHP Frameworks Advantage and Disadvantage*. [online]. Smart Mentors. 2018, [cit. 2021-12-29].
- [40] TRIPATHI, A. *Best Front End Frameworks for Web Development of 2021: The Complete Guide* [online]. Geek Culture, 2021.
- [41] WEIZENBAUM, N. *Ruby Sass Has Reached End-Of-Life* [online]. 2019.
- [42] ZEMČÍK, P. *Tvorba uživatelských rozhraní, studijní opora*. 2006. s. 71.
- [43] ČESKÁ NÁRODNÍ RADA. *Listina základních práv a svobod*. 1992. článek 17 (2), článek 19 (1).